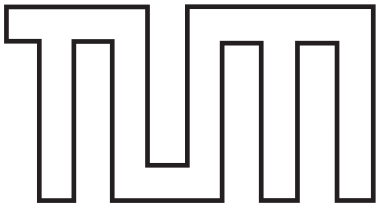
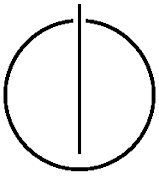


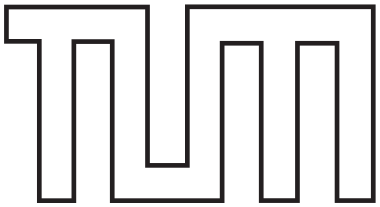
**Department of Informatics**
of the Technische Universität München

BACHELOR'S THESIS IN INFORMATICS

# Development of a Social Extension for Real-Time Communication in CAD Software

BY MARKUS MÜLLER





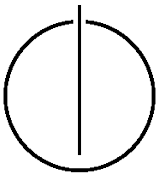
**Department of Informatics**
of the Technische Universität München

BACHELOR'S THESIS IN INFORMATICS

# Development of a Social Extension for Real-Time Communication in CAD Software

# Entwicklung einer sozialen Erweiterung für die Echtzeitkommunikation in CAD Software

AUTHOR: MARKUS MÜLLER
SUPERVISOR: PROF. DR. FLORIAN MATTHES
ADVISOR: MARIN ZEC, M. SC. WITH HONORS
SUBMISSION DATE: OCTOBER 15TH, 2015



## ASSERTION

I confirm that this bachelor's thesis is my own work and I have documented all sources and materials used.

Markus Müller

## ABSTRACT

Social Software offers great benefits for businesses. In this thesis, we analyze how instant messaging can be integrated into the CAD application Solid Edge and if users see a need for such functionality at their workplace. We took a look at existing instant messaging products, conducted a short literature review of tries of bringing chat into the workplace and interviewed 12 CAD users on how they currently solve problems. The thesis describes the requirements for an integrated instant messenger and details on its implementation. The resulting "EmbeddedChat" was evaluated in an online survey with 91 participating Solid Edge users. We conclude that a high portion of participants has reservations about the introduction of an integrated instant messenger at their workplace.



KEYWORDS: Social Software, Instant Messaging, Work Chat, Collaboration, Computer-Mediated Communication, Group Support Systems, Web Application Development

RESEARCH AREAS: Information Systems, Computer Science, Computer-Supported Cooperative Work (CSCW), Computer Aided Design (CAD)

CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# 1 INTRODUCTION

Since people have thought of personal computers, they thought of using them for facilitating group work and collaboration. A popular example can be found in the article "As We May Think", where Bush (1945) describes a device called *memex*, which "today we might call the personal computer".

Brought into common usage in 2002 by Clay Shirky who organized a *Social Software Summit*, *Social Software* is now the main term to describe this type of software (Allen, 2004). Popular Social Software includes Social Networks, Blogs, Wikis, Discussion Boards and Instant Messaging amongst others (Bächle, 2006). Nowadays, both private and business matters benefit greatly from Social Software - we will mostly focus on the business side during this thesis.

This thesis is part of a research cooperation between our chair at TU München, "Software Engineering for Business Information Systems (sebis)", and "Siemens AG Corporate Technology". Goal of this cooperation is to examine how certain types of Social Software can be integrated into existing software and which additional use they provide for their users. Because *existing software* is a term seized too broadly, Siemens' 3D CAD (Computer Aided Design) tool *Solid Edge* was exemplarily selected at the beginning of research. After the previous master's thesis of Gleixner (2015) "Implementation of Collaboration Features in CAD Software" coped with the implementation and evaluation of an integrated *Question&Answer system Social Edge*, this thesis focuses on *Instant Messaging*. We propose our integrated instant messenger *EmbeddedChat*.

For this thesis, we determined three main research questions:

- How do CAD designers communicate in their daily work?
- How can an instant messaging client make use of a CAD programs' context?
- Do CAD designers find instant messaging an useful addition to their job?

To gain knowledge about the topic, we analyzed current existing instant messaging products and conducted a literature review. The results can be found in section 3. Qualitative interviews with Solid Edge users were then conducted to answer the first research questions and sharpen the requirements for an integrated work-chat (see section 5). By developing *EmbeddedChat*, we could answer the second question (see chapter 4). In chapter 7, we examine the last research question through an online-survey with 91 Solid Edge users.

# 2

# RESEARCH APPROACH

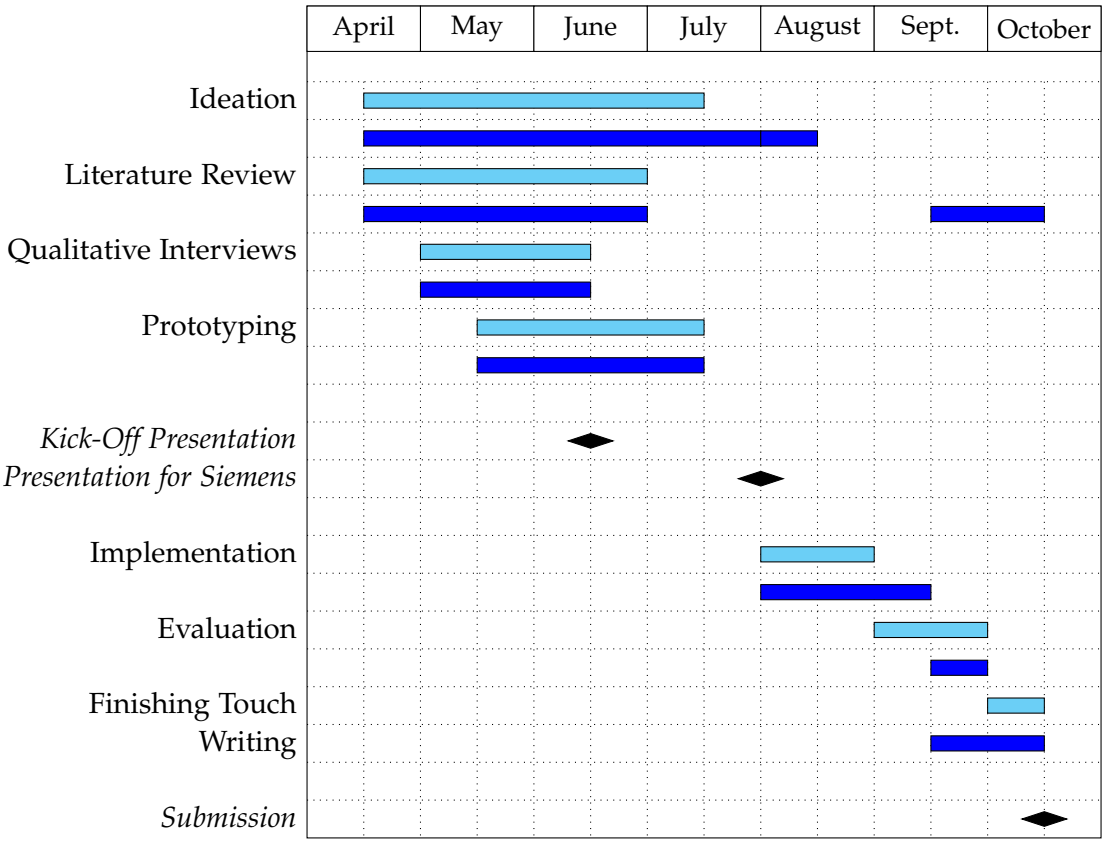



Figure 1: Planned (light-blue) vs. Actual Time (blue) Schedule. Milestones are noted as Diamonds.

Work on this thesis began in the mid of April. We directly started with an ideation phase on how the planned Social Extension for real-time communication in Solid Edge should look like. Parallel to that, we reviewed existing instant messaging applications and research papers on work-chats in business areas. Applications were chosen according to their general popularity (WhatsApp, Facebook Messenger, Chat in Google Docs), recent popularity (Slack) and similarity in the area of application (Communicator for Revit) and analyzed in terms of functionality and use cases. In order to find work-chat-related research papers, we mainly used Google Scholar. At the end of work, we needed to repeat some parts of our analysis because we missed on taking enough notes in the first phase of literature review. You can find the results in chapter 3.

After developing an initial concept for our extension, an escalating process which is focused on problem solving, we conducted several qualitative, open, interviews with Solid Edge professionals and students who participated in the competition "F1 in Schools"

(section 4.2). Through existing Siemens contacts and own experience in F1 in Schools, we could establish these contacts quite easily and could meet our time expectations.

Since mid-May we experimented with implementation approaches - mainly to find out and evaluate which features were possible to implement in the given time. Features included voice and web-cam chat and screen-sharing capabilities, but were given up because of technical difficulties with the Solid Edge API. We presented our state of ideas in a kickoff presentation at our chair on June 15th, later, on July 29th, in another presentation in front of the Siemens executive responsible for the research cooperation between Siemens and our chair.

Because we experienced difficulties with our first concept, partly because of the qualitative interview's results, we had to extend our ideation phase into mid of August and have it overlapping with the implementation phase. We therefore started work first on implementation parts which were independent of the chosen concept (e.g. named as a requirement in any way). Implementation was then finished later than planned, in mid-September; details can be found in chapter 6.

To evaluate the final EmbeddedChat, we conducted an online survey which contained a 2:22 minutes video of EmbeddedChat in use. To evaluate the perceived usefulness, Davis (1989) was consulted. The survey was available in English and German and ran two weeks, from September 21$^{st}$ to October 2$^{nd}$, and attracted 91 Solid Edge users as participants. You can find information about survey design, results, discussions and limitations in chapter 7.

# 3

# RELATED WORK

Instant Messengers are currently being used by over 25 million people in Germany (see figure 2) and are therefore one of the most important types of Social Software.

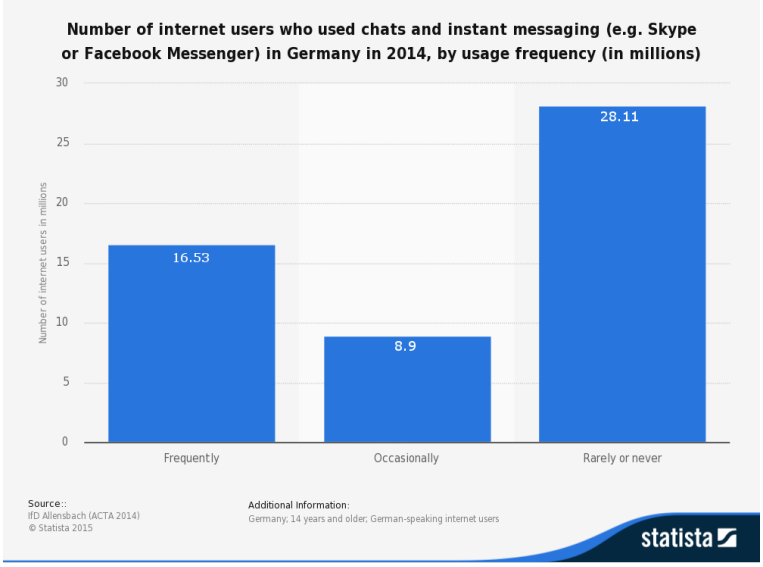



Figure 2: Number of Internet Users who used Chats and Instant Messaging in Germany in 2014, Statista (2015b)

After defininining the most important terms, we took a look at existing instant messaging products to have a solid knowledge base for designing our *EmbeddedChat* in sections 4 and 5. We analyzed generic products, which are rather targeted on private use, as well as productive and business-specific products. We then followed with a short analysis of research results in bringing chat to the workplace (section 3.3).

## 3.1 DEFINITIONS

In order to talk about certain topics, we first need to define them.

CHAT AND INSTANT MESSAGING Oxford Dictionary describes "chat" as an "online exchange of messages in real time with one or more simultaneous users of a computer network". A clear distinction to "Instant Messaging (IM)" is hard - in literature, both terms are often used interchangeable. Two differences are sometimes made: Firstly, when talking about a "chat", the underlaying program is not as important as when talking about "IM". Secondly, IM often only describes the message exchange with one person (Nardi et al., 2000, p. 80), whereas "chat" specifically is targeted onto group conversations.

*PC.net* (Christensson, 17.9.2004) makes a quite strong distinction which cannot be found in research this definite.

GROUPWARE AND COMPUTER-SUPPORTED COLLABORATIVE WORK (CSCW) Allen (2004) cites the definition of "groupware" from Peter and Trudy Johnson-Lenz in 1978: "Intentional group processes plus software to support them". As he further elaborates, the new term "Computer-Supported Collaborative (or sometimes Cooperative) Work (CSCW)" was brought in by the academic community in 1984 because they were not happy with either the term "office automation" or "groupware" for research into how groups use computers to collaborate. The main difference definitions show is that groupware rather describes the technology whereas CSCW specifies the field of studies.

COMPUTER-AIDED DESIGN (CAD) WITH SOLID EDGE Computer-Aided Design software can be used to create two-dimensional (2D) drawings or three-dimensional (3D) models (Rouse, 2011). Solid Edge is a CAD program for Microsoft Windows by Siemens Product Lifecycle Management (PLM) Software Inc. (before 2007 known as Unigraphics Solutions, UGS). The first version was released in 1995 (Wikipedia, 2015c), the current version is ST8 which was published in June 2015 (Zwettler, 10.06.2015). A screenshot of Solid Edge in action is shown in figure 3

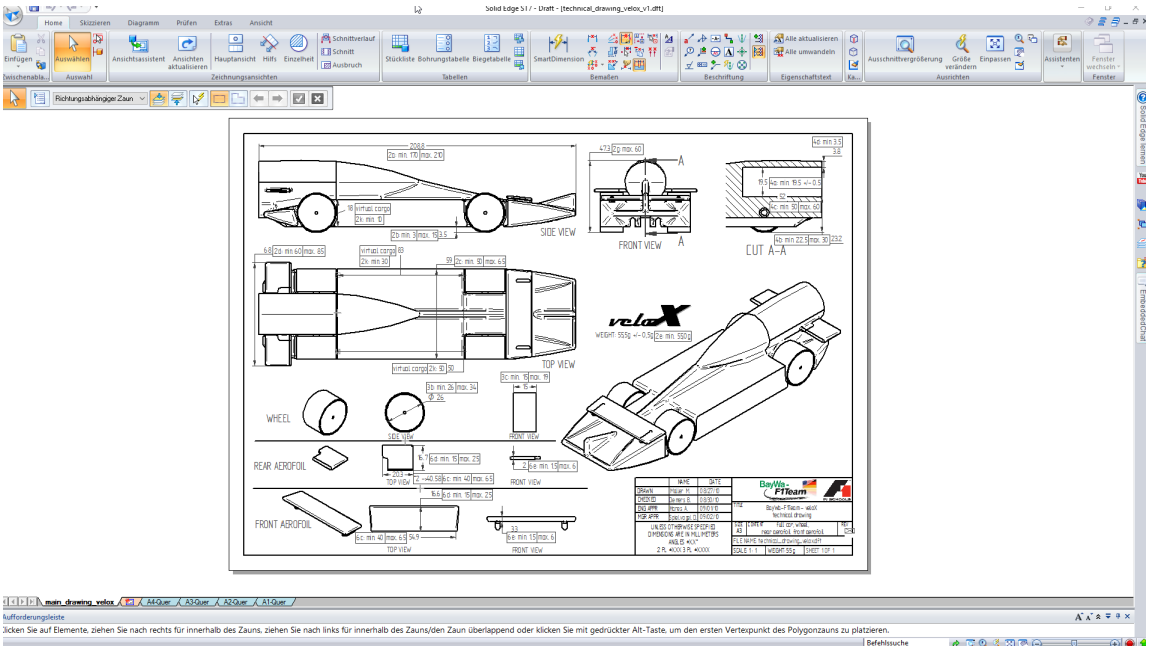



Figure 3: Screenshot of Working on a Technical Drawing in Solid Edge

F1 IN SCHOOLS "F1 in Schools" (F1 in Schools) is an international student competition for students between 11 and 19 years. In teams of 3-6 members, students design and build a miniature formula 1 car and compete against other teams from all over the world - first in regional, then in national competitions and then finally in the World Finals (2015 in Singapore). Teams have to use professional tools to succeed. In Germany, Siemens is a

sponsor of F1 in Schools and offers student free versions of Solid Edge. Because these students mostly started to work with Solid Edge just months before the competition and therefore have an especially great need for information on how to use the program correctly in order to achieve good construction results. So they are an interesting group to examine as can be seen in section 4.2.

## 3.2 EXISTING CHAT APPLICATIONS

### 3.2.1 *Chat Applications for Private Use*

For analyzing chat applications which are most used for private matters, we picked the two most-used applications (Statista, 2015c): WhatsApp and Facebook Messenger.

#### 3.2.1.1 *WhatsApp*

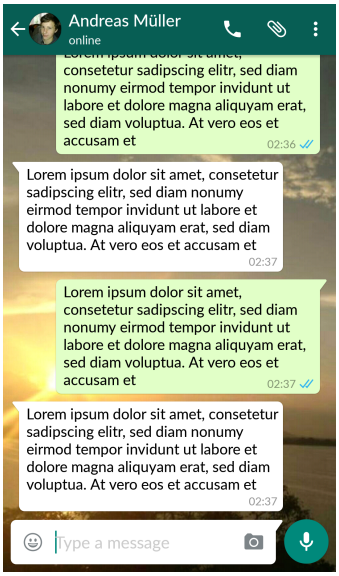



Figure 4: Screenshot of a WhatsApp Chat

WhatsApp is one of the most used mobile Instant Messengers world-wide, especially in Europe. In Germany, about 57% of all mobile internet users are using WhatsApp, in Spain usage is as high as 70% (Statista, 2015d). Founded in 2009 by Jan Koum and Brian Acton and marketed as a cheap and simple SMS replacement (McMillan, 20.02.2014), its user base has steadily grown to now about 900 million since then (Guynn, 2015). Generally, its price model is 0.89$/€ per year for most systems, while iPhone users only have to pay a one-time fee. WhatsApp gained a lot of media traction after being taken over by Facebook for an unprecedented sum of 19 billion US-dollars in 2014 - the biggest acquisition price ever paid for an internet startup (Olson, 2014).

KEY FEATURES One of the key features of WhatsApp is its simple and clean user interface (see figure 4). Users can use instant messaging and group chat up to 100 members. Within a chat, photos, videos, audio, location and contacts can be sent to the other participants. Because the phone address book gets synchronized with WhatsApps' servers, users easily see who of their contacts uses WhatsApp, too. Additionally, they can see the last time their contact has been online. An advantage of WhatsApp compared to other mobile instant messengers is its compatibility with different smartphone OS versions, even old ones like Nokia Symbian OS (source: WhatsApp). In general, WhatsApp is *mobile-only*, meaning it only exists as an app for smartphones, registration is only possible with a valid mobile phone number. Even tablets with the same OS are not supported for app installation. A desktop web-version was published in January 2015 (WhatsApp, 2015), but requires an active internet connection on the smartphone. Since March 2015, a VoIP (Voice over IP) call functionality is also available (Geiger, 2015).

The address book synchronization has not only received positive reception (Wisniewski, 2013). With all contacts relationships being stored on WhatsApp's servers, it could lead to a giant web of relations WhatsApp could make use of. In combination with the acquisition by Facebook, a company which makes most of its revenue in advertising, privacy concerns have been increasing (Page, 2014). As an additional privacy concern, it has been criticized that WhatsApp does not make it clear how messages are encrypted. WhatsApp's source code is closed source, so an independent security check can not be easily conducted (Stack Exchange, 2015).

USE CASES Because of its focus on mobile devices and simple interface, WhatsApp serves as a replacement for SMS which only requires mobile data. Deriving from our own experience, WhatsApp is used for everyday personal communication. Because a user keeps his smartphone with himself most of the time, he is available all the time and can answer messages on the go.

FEATURE SUMMARY

- Simple User Interface
- Instant Messaging
- Group Chat
- Media Sending
  - Images & Photos
  - Videos
  - Audio
  - Location
  - Contact Data
- Automatic Contacts List from Phone Contacts
- VoIP Calls

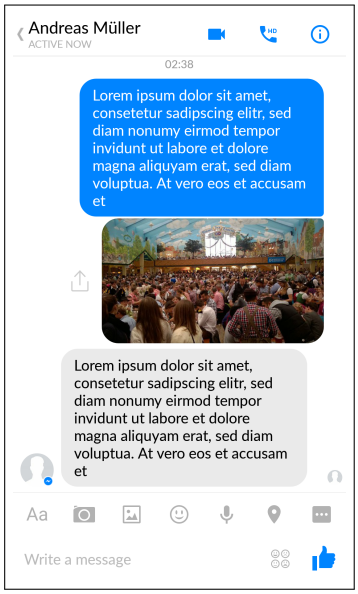



Figure 5: Screenshot of a Messenger Chat

#### 3.2.1.2 *Facebook Messenger*

Facebook Messenger (often just called *Messenger*) is the name of chat functionality integrated into the Facebook social network, the largest social network worldwide, founded in 2004 (Phillips, 2007). Facebook Messenger has about 700 million monthly active users (Statista, 2015a), about the half of the monthly active Facebook users (1.44 billion, see PR Newswire (2015)). The whole service is, as it is Facebook, completely free - users *pay* with their data so personalized advertisements can be shown in Facebook's applications.

KEY FEATURES Messenger has basically the same functionality as WhatsApp. As an addition, stickers can be sent to conversation partners. Stickers are images which are organized in packs. Each pack is devoted to a certain theme. Example sticker packs would be *Cat Stickers* or stickers of a brand like *Minions* (see figure 6). However, Messenger's main features are probably the deep integration into Facebook - all of your Facebook contacts are automatically accessible via Messenger - and its availability on most different device platforms, not only smartphones. Interesting is its split in two different styles: On the one hand, it's integrated into the Facebook website and therefore adapts the website's style. On the other hand, it maintains a different branding in *dedicated applications*. This separate branding seems to be more modern and adapted to current mobile design standards. Examples are the dedicated web-application *messenger.com* and apps for mobile devices.

Messenger's newest feature was introduced in March 2015 (Constine, 2015) - *Apps for Messenger* enables third-party iOS and Android apps to integrate itself into the Messenger app. So, especially enriched sounds or videos can be sent to a conversation. An example: *ClipDis for Messenger* allows users to "turn simple text messages into video mashups" (ClipDis) and send the final video via Messenger.

Figure 6: A Sticker available for Facebook Messenger

USE CASES Messenger can be used wherever the user stays and no matter between which devices the user switches. With its dedicated apps, not even a Facebook account is necessary to use Messenger.

FEATURE SUMMARY

- Available for Almost Every Device Class (Desktop, Smartphone, Tablet,...)
- Instant Messaging
- Group Chat
- Media Sending
  - Every file type with integrated image, audio and video displaying functionality
  - Location
  - Stickers
- Automatic Contacts List from Facebook Contacts
- VoIP Calls
- Third-Party Apps for enriched messages supported

### 3.2.2 *Chat Applications for Business-Related and Productive Use*

#### 3.2.2.1 *Google Docs Integrated Group Chat*

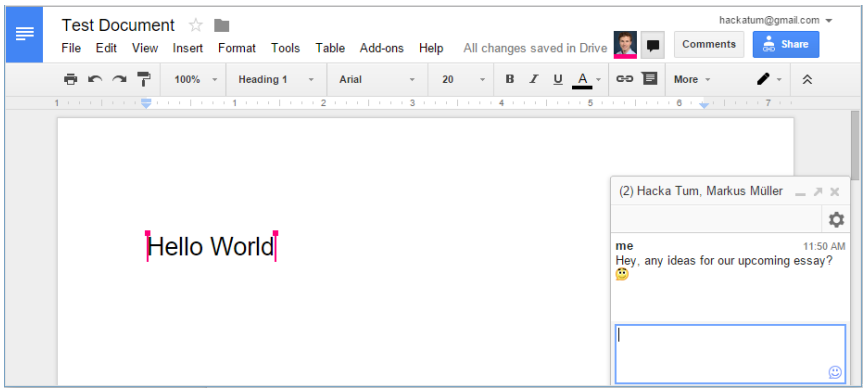



Figure 7: Google Docs with Group Chat in the bottom right corner

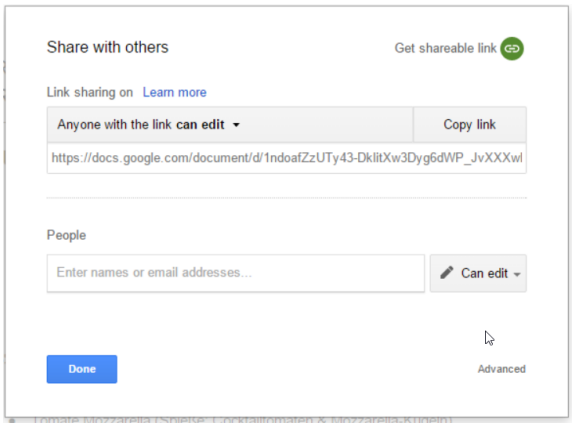



Figure 8: Google Docs Share Document View

Google Docs, introduced in 2007 (Wikipedia, 2015b) and now part of Google Drive, is an office applications collection which people can use to create text documents, spreadsheets and presentations amongst other data types. It is free for individuals, fees for business use start at 4€ per user per month (Google Apps for Work). In it's desktop version, it is a typical example for a web application - users can use Google Docs right in their web browser. A separate desktop application does not exist. Therefore, working on documents can be continued on arbitrary computers. On mobile devices, document contents can be viewed within the browser, but dedicated apps need to be installed for full editing functionality.

KEY FEATURES Besides the web application aspects, a main feature of Google Docs is its collaboration functionality. Users can *share* their documents and work on them *simultaneously* with other users. For the share screen see figure 8. All document changes are recorded and can be viewed in a history of past revisions which can be restored if needed.

If collaborators are viewing a document at the same time, *group chat* can be used. Group chat happens within an integrated view on the bottom right corner (see figure 8) and does not have persistent memory - a chat member only sees messages which were sent after him joining the conversation.

USE CASE Summing up, Google Docs offers a powerful way to work on various documents with remote group members or co-workers. We identify the integrated *group chat side bar* as a simple and useful tool to discuss open questions or decisions without having to leave the *current work context*.

FEATURE SUMMARY

- Integrated into Google Docs Document View and Edit User interface
- Group Chat with Online Collaborators
- Document Changes are Propagated in Real-Time
- History of Past Revisions

#### 3.2.2.2 *Slack*

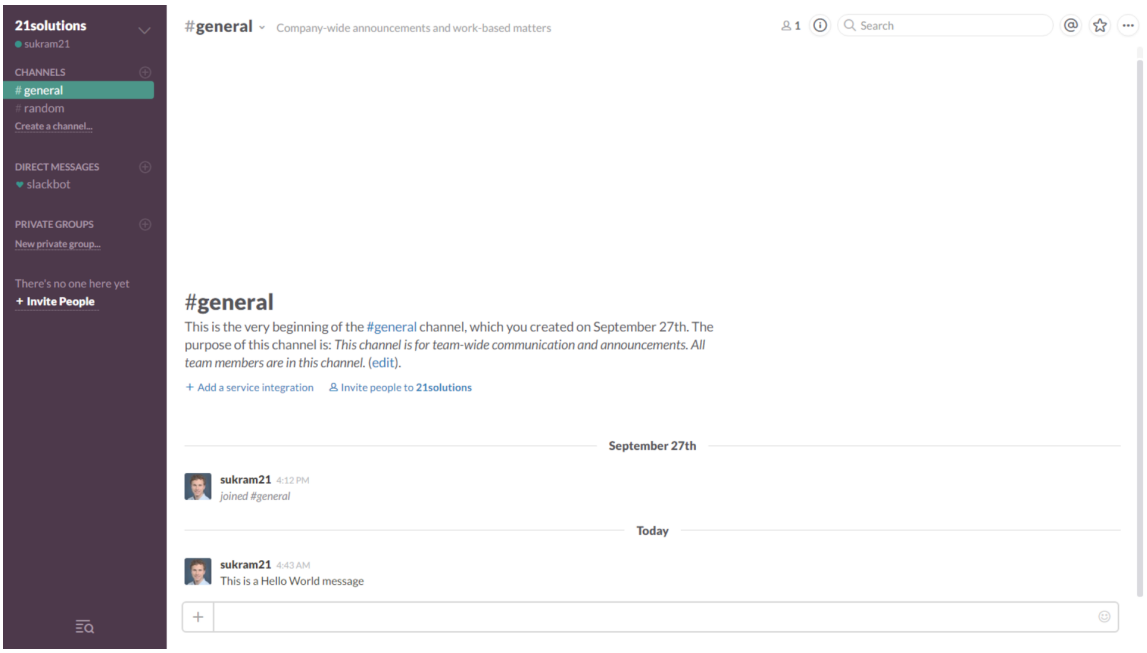



Figure 9: Slack Chat Interface

Founded in 2013 (Koetsier, 2013), Slack is a very new team and business chat, but gained a lot of attention by getting over one million users out of which 300.000 are paying, until June 15 (Newton, 2015). Slack's goal is to become the center of team communication ("Team communication for the 21st century", Slack (a)) and to reduce the number of e-mails sent to a minimum. Slack is mainly a web application with native applications for all major desktop and mobile OS. There exist several reports of teams who have fallen "in love" with Slack (York, 2015; Crew). The main functions in Slack are free for an unlimited number of users, limitations apply mainly to the number of recent messages included in the search (free: 10.000) and file storage (free: 5GB). Paid plans with extended functionality start at 6.67$ per user per month (Slack, d).

KEY FEATURES In Slack, communication is built around *channels*, another word for *chat rooms*. Channels are written with a prefix #. Users are free about which channels they want to create - Slack gives examples (Slack, b) which include *Team Channels* (e.g. #engineering), *Topical Channels* (e.g. #lunch) and *Location-Based Channels* (e.g. #sf, #nyc). All members of an organization can view all contents of a channel and can join or get added to channels to participate in the discussion. To have private conversations, private groups and direct messages (instant messaging) are possible.

Some of Slack's features: Within conversations, users can post arbitrary files and images. Links to popular services like YouTube are automatically expanded to display an inline preview. Messages can be formatted using Markdown syntax. Users can be mentioned within a message and will receive a notification.

Slack emphasizes its search functionality which understands several filter parameters like "during:" to find messages within a certain time period. Besides text messages, all uploaded files and link meta data are searchable with the built-in search. For links to files on services like Dropbox and Google Drive, the corresponding file contents get indexed - the search index is auto-updated on changes.

Slack offers integration with over 50 external tools, mostly for getting notifications from these services inside the channels. When using the GitHub integration for example, new commits, pull requests and activity on GitHub issues will be posted to a channel and therefore get indexed by Slack's internal search. More integrations can be added manually.

USE CASES Because it is a relatively new tool, Slack is mainly popular among tech companies, software developers and startups. This also can be seen when looking at the list of existing integrations with external services (Slack, c). These integrations seem to be one reason why Slack got successful. Slack excels at bringing context information from different applications into one central platform where all information can be viewed and searched.

FEATURE SUMMARY

- Available for Almost Every Device Class (Desktop, Smartphone, Tablet,...)
- Instant Messaging
- Group Chat
- Media Sending: Every file type with integrated image and video displaying functionality
- VoIP Calls
- Integration with many external tools, mainly for notification purposes
- Powerful search with file content indexing

#### 3.2.2.3 *Autodesk A360 Collaboration for Revit - Communicator*

Autodesk A360 is a web-based cloud platform for project-based collaboration using Autodesk CAD-products. Autodesk is one of the largest CAD software companies worldwide. People can use A360 to upload documents and CAD files, view them within the browser window, post activity updates and comment on them. There exist several apps for mobile devices to get access to A360 (see Autodesk).

*Revit*, a desktop software used for planning and designing of buildings, and in general "Building Information Modeling (BIM)", is on of the Autodesk CAD tools which can be enriched with A360 functionality. This functionality is called "Autodesk A360 Collaboration for Revit" and was released in the end of 2014 (White, 2014). An important part we will now examine is the *Communicator*.

KEY FEATURES The Communicator is integrated in Revit as a sidebar (see figure 10), but can be undocked into a separate window. At first, it offers a single group chat with

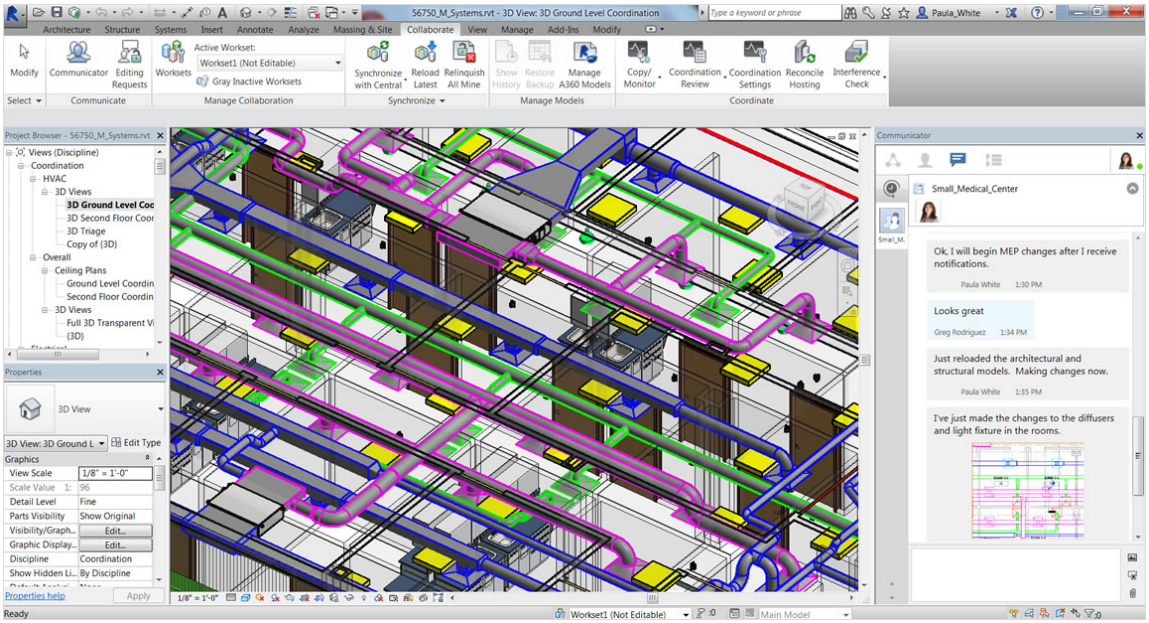



Figure 10: Revit Communicator integrated as Sidebar; screenshot from Khemlani (2014)

all members involved in the project. One to one conversations are possible, too. When chatting, users can send images and files. Worth noting is the integrated possibility to send a screen shot, where the user can select the area of the screen he wants to capture. Export of chats (most recent 3.000 messages) is also possible.

Grouped under the title "Contextual Collaboration Tools", users can, amongst other things, view on which model their project partners are working on right now and view past revisions of a model. This so called *Activity Feed*, the list of past revisions, is possible because in A360, models are stored centrally on a server and synchronized after changes made (Knittle, 2015; Autodesk, 2015; Autodesk Building Solutions, 2015).

USE CASES As we have seen, the Communicator in A360 Collaboration for Revit offers a way for CAD designers and project members to communicate without leaving their CAD-program context. Additionally, different types of context information are provided, which otherwise one would have to ask. Even if our ideas for *EmbeddedChat* mostly happened before we discovered Communicator, we see it as a role model and note that supporting collaboration via an instant messenger is already tried in the CAD industry.

FEATURE SUMMARY

- Integrated into Autodesk CAD software Revit as a sidebar or as a separate window
- Connected with Autodesk Cloud Project-Platform A360
- Instant Messaging
- Group Chat, but only with the whole project group
- Media Sending
  - Images and Files
  - Screenshots with possible selection of area to capture

- Context Information
  - View on which model project partners are currently working
  - View revision of a CAD model

## 3.3 RESEARCH RESULTS IN BRINGING CHAT TO THE WORKPLACE

We mainly looked for research publications which address the introduction of instant messaging to businesses. By that, we hoped to gain insight about challenges and methods to take into consideration, in order to develop our own business instant messaging client. We ended up with three papers: Bradner et al. (1999); Herbsleb et al. (2002); Muller et al. (2003). With the most recent paper being from 2003, it is a pity that newer papers regarding this topic do not really exist, especially because a lot of development happened within the past years. Out of the existing chat applications presented in the last section, the Facebook chat system is probably the oldest one with Facebook having launched in February 2004 - Slack, the youngest one, but already popular, launched in August 2013.

Bradner et al. (1999) was one of the first papers who "look[ed] at the use of chat in business settings" (Bradner et al., 1999, p. 140). They introduced a group chat called "*BABBLE*" to six groups at IBM Corporation and examined adoption and use within the 6-month field study. They experienced very different adoption rates among the groups and discussed three concepts which they found useful in trying to understand adoption (Bradner et al., 1999, p. 152): Critical mass, social affordances and interaction ecologies. (1) In terms of critical mass, they discovered that the composition of participants had a higher impact on reinforced use than the number of people involved (Bradner et al., 1999, p. 152-153). (2) When looking at a specific feature or use case, they realized that different people see the feature or use case as different socially affordable. (3) They began to compare chats like BABBLE to "ecosystems populated by communicative practices" - they required the input of human participants to be be kept alive. For our research, we conclude that finding the right group of people for a real testing phase is extremely important, more important than the pure number.

Herbsleb et al. (2002) developed and deployed their tool "Rear View Mirror (RVM)" to facilitate multi-site software development work. One main issue they wanted to solve was the virtual absence of informal communication, so called "corridor" or "water cooler" talk, between sites. Main features were a presence viewer, one-to-one instant messaging and group chat. Among their main findings were that introduction of RVM was more difficult than expected. One reason was that IM and chat were perceived rather "superfluous" and fear of being distracted was high. Another reason was that there was no clear, well-defined need for IM - and "water cooler talk", whose absence they wanted to solve, was not seen as desirable, because it is not seen as "real work". The third reason was that team members of remote sites were often seen as hostile because they were not familiar with each other. So an increase of communication with those seemingly unfriendly people was often not wanted. We conclude that chats and the "water cooler talk" they generate are often recognized as a source of distraction. This may be true on excessive use, but generally, "water cooler talk" is an important part in daily work-routine. Employees should be able to opt out of a chat anytime to lower their fear of getting dis-

tracted. It would be interesting to observe if Solid Edge users have the same reservations against instant messaging.

After the introduction of Lotus Sametime (IBM communication solution with instant messaging capabilities) into three business organizations, Muller et al. (2003) conducted a detailed study of the maturation of IM at one company and deduced an "Instant Messaging Maturity Model". The "Early Stage" is characterized by easy adaption of the technology, but yet mainly communication with well-known team members and friends. In "Maturity Stage", users begin to improve their chat behavior and start finding new reasons to use IM as a necessary tool. The "Later Stage" is only hypothesized - Muller et al. (2003) expect users to start managing IM use in terms of interruptions and privacy because of grown contact lists and large conversations which would, without intentional management, become a distraction from the job. In general, the "Instant Messaging Maturity Model" means for us that a real testing phase will require a certain amount of time to generate valid results. If conducted within a short time period, user feedback would probably be notably different from feedback on long-term usage.

# 4

# CONCEPTUAL DESIGN

At the start of work on this thesis, we brainstormed on features a *Social Extension for Real-Time Communication* in Solid Edge should have. We then talked with professional Solid Edge users and students who discovered Solid Edge through the international competition *F1 in Schools* (see section 3.1). We developed a first concept - centered on a process for solving problems together. We later did not implement this concept because we realized that the process would have gotten too complex with too many questions unsolved. Instead we decided to implement a more generic and flexible version.

## 4.1 FIRST CONCEPT

We drew the following scenario: A user A is constructing a CAD model with Solid Edge or using Solid Edge for another task. He realizes that he does not know how to achieve a certain result with his model. He needs to talk to someone with more knowledge in this area to solve his problem as fast as possible and in order to stay productive. In his direct surroundings, there is no one who could help him.

In our first brainstormings, we then determined these first ideas for a *Social Extension for Real-Time Communication* in Solid Edge:

INSTANT MESSAGING As already stated, the main goal of this work was to implement an instant messenger for Solid Edge users. Users should be able to contact other users from their contact list and send text messages. These messages should be delivered in real-time with the other user to be notified if he was online. Even if communication where both contacts are online at the same time was the use-case we most focused on, it should also be possible to send messages to offline contacts. We decided to focus first on one-to-one chats instead of group chats because complexity might be too high and problem solution works best with a concrete contact person.

INTEGRATION INTO SOLID EDGE The extension should integrate into the Solid Edge program and user interface so users do not need to switch program contexts between working and communicating. We have already seen integrations of instant messengers into applications for Google Docs (3.2.2.1) and Revit - Communicator (3.2.2.3) and will use them as role models.

MAY DAY BUTTON FOR INITIATION OF A CHAT Deriving from *Amazon Mayday Button*, when a user has a problem or difficulty, he should ask for help with one click. With the Amazon Mayday button, Fire Tablet users can call one of Amazon's technical experts at any time (Amazon.de). We would like our extension to offer such a button and get in contact with another user who can offer support as fast as possible.

SCREENSHOTS Within a conversation, it should be easily possible for users to send screenshots of their current screen state. By "easy" we mean, there should be a button to capture and send the current screen with one click.

EVENT QUEUE/TIMELINE To get as much information as possible about the problem context, conversation partners should see the actions and commands a user executed while having the conversation open. The questioning user so would not have to describe the actions he already tried in written form and therefore loose time, but could just redo his previous actions. The transcript of commands is automatically sent to the other participant.

## 4.2 QUALITATIVE INTERVIEWS

To confirm our use case and generally to learn more about how communication is done by Solid Edge users, we conducted several qualitative interviews with professional Solid Edge users and students who discovered Solid Edge through the international competition F1 in Schools. By qualitative interviews, we mean open interviews with no fixed course of conversation or fixed questions (King et al., 1994). The exact answers can be found in appendix 9.1.

| **Name** | **Company 1** | **Company 2** | **Company 3** | **F1 in Schools Participants** |
|---|---|---|---|---|
| Number of Employees | 15 | 165 at visited site | 60-100 | 3-6 students per team |
| Number of Interviewed Persons | 1 | 4 | 1 | 6 |
| Age (estimated) | 40-49 | 50-59 | 25-39 | under 18 |

Table 1: People we interviewed in our Qualitative Interviews

Through one Siemens Solid Edge sales partners, we initiated contacts to three companies of which we interviewed six employees who use Solid Edge on a daily base. In one company, the interviewed person was the only one who regularly used Solid Edge, in another company, the interviewed person had one co-worker who also used Solid Edge. The third company had the biggest userbase - we interviewed four CAD designers. For the F1 in Schools competition, we visited the German Championship and conducted interviews with six constructors from four teams.

### 4.2.1 *Main Interview Structure*

| | |
|---|---|
| *to F1* | How did you acquire your Solid Edge knowledge? |
| *to all* | How do you currently solve problems or issues related to Solid Edge? |
| *to all* | How do you communicate solving these problems? |
| *to all* | What is your opinion on an integrated instant messenger for Solid Edge (with our first ideas as features)? |

Table 2: Interview Structure for Qualitative Interviews

In table 2 you can see the rough structure of our qualitative interviews. From F1 in Schools participants, we first wanted to know how they got their knowledge with Solid Edge. Because F1 in Schools participants normally do not have a Solid Edge community nearby where they can learn using CAD modeling, they then automatically need to think of good communication methods to have contact with people who can support them. The next two questions were directed on how Solid Edge users solve problems and issues and mainly, which means of communication they use for that. In the last part of our qualitative interviews, we presented our ideas of an integrated chat extension from 4.1 and asked for general feedback.

### 4.2.2 *Learnings*

*Solid Edge users use the internet to find solutions, but very passively*

We discovered that all interviewed Solid Edge users are searching in the internet for problem solutions. They mainly use search engines like Google and popular Solid Edge discussion boards like German CAD.de and English Solid Edge Forum. Use of the internet happens only passively, no one of the interviewed users has asked for help actively in the internet, e.g. in a discussion board.

*Communication in Person is preferred*

Not very surprising, all persons we talked to found solving their problems with another person the most useful and used all occasions to meet their colleagues instead of having to contact them via media.

*Communication with (familiar) Team Members is preferred, support later*

Professional Solid Edge users with a support contract with a sales and support company try to solve their difficulties mainly on their own before contacting the support. They even invest hours of time to solve the problem and see contacting the support as a last resort.

*No instant messenger use by professionals*

No one of the six professionals interviewed uses an instant messenger within their daily work. They mostly rely on phone communication and e-mail.

*F1 in Schools users try to use tools they use for private matters and fail*

We discovered that students tried to use the mobile instant messenger WhatsApp, a very familiar instant messenger for them, to solve their desktop Solid Edge problems. As they admitted, this resulted often in time-consuming description text writing and photographs of their screen.

*No direct denial of a chat, but no clear need and euphoria*

All interviewed people were not averse to an instant messaging extension like proposed, but also not very euphoric and did not express a need for it.

### 4.2.3 *Limitations*

The main limitation concerning our interviews is that the biggest group of interviewed people is from the same company (company 3) and does not really have the challenge of distributed workplaces. They are mostly sitting in the same room with the rest of their CAD department located in a room close by. As a result, the most named communication method was to meet in person. We could not learn as much about communication with distant colleagues as we hoped to.

Another limitation is the sample size - with three companies and four student teams, we probably could not capture the whole diversity of communication methods used by Solid Edge users.

In the end, we have to say that these qualitative interviews indeed helped us to understand Solid Edge users better and familiarize with them, but were not that gainful to improve concrete parts on our chat concept.

## 4.3 FIRST CONCEPT'S PROCESS

Our first concept was centered on solving a problem fast and was similar to seeking for support in a Q&A board (see Gleixner (2015)). Users file a help request and get help by experienced users - the solution can be viewed by all users of the platform and helps them in case the same problem occurs for them.

According to our process, when a user needs support, he clicks the *Mayday* button and then can choose between three alternatives. The alternatives form an *escalating problem solving process*. They are ordered according to the time it takes to solve the problem using this particular alternative and behave like steps - if the first alternative did not work, the

second and then the third one should be chosen. First, a user can search for a solved solution within a database of public archived conversations. If no suitable solution is found, he can *file a help request* (second alternative). He enters a short description of the problem as new conversation title and a proper online user with knowledge in this area is suggested for chat. If no suitable user can be found, the questioning user can choose a *contact from his existing contact list* (third alternative).

After a conversation, the user can mark it as finished and add the status of *solved* or *not solved* and a short summary. The final conversation can be saved publicly (default) and other users can access it by using the first alternative.

DISCARD OF FIRST CONCEPT In the end, we did not go along with this concept because of several reasons. Several questions remained unsolved:

- Because conversations are bound to problems, a user could theoretically have multiple open discussions with another user. Will this not cause confusion?
- If a conversation contains messages which should not be published (e.g. informal, not problem related), how can these messages removed so the whole conversation can be publicly published? Will users trust the removal and publish a conversation which former included such messages?
- What happens if one conversation participant wants to publish a conversation publicly, but the other one does not?
- How can informal, water-cooler talk, which is one central advantage of instant messaging (Herbsleb et al., 2002) get implemented in this process which is optimized for solving problems?
- Is this relatively fixed process flexible enough to fit the needs of users? Will users adapt to the concept?

Because solving all these open questions would have resulted in an application too complex for a bachelor's thesis, we decided to reduce our concept to a more generic and flexible one. It corresponds more to classic instant messengers which can be an advantage because it is easier to understand.

## 4.4 SECOND CONCEPT

In our new concept, we went away from a problem-centric approach to a more generic one. Users should use an instant messenger the way they like to. We omitted the first two alternatives - a user can now only choose contacts out of his contact list to communicate with and have one only one open conversation with another user at a time. Within a conversation, we do not dictate about what participants should talk about. We keep an *Archive* feature, but this time, the conversation is not published but only saved for the involved users.

The result is an integrated instant messenger *EmbeddedChat* enriched with context information.

# 5

# REQUIREMENTS

From our second concept, we deducted these functional and non-functional requirements.

## 5.1 FUNCTIONAL REQUIREMENTS

CORE INSTANT MESSENGER FUNCTIONALITY A user should be able to conduct conversations with another user. Within a conversation, it should be possible to send *text messages* and *images*.

DIRECT SCREENSHOT SENDING The user should be able to capture and send a screenshot of his whole main screen with a single button. The screenshot is treated as a normal image.

ACTIVITY CONTROL: EVENT TIMELINE When a user uses a command while viewing a conversation, a textual representation of this command is sent to a special *Event Timeline* which can be viewed by all conversation participants. It should be possible to deactivate this feature to prevent privacy concerns.

GROUP CHAT It should be possible to create conversations with more than two participants. A user who creates a group conversation should be able to choose participants out of his contacts and to set a title. All participants have the same access rights and can view all messages and command events within the conversation. To distinguish the messages between each user, the user's profile picture should be shown next to the message.

INTEGRATION INTO SOLID EDGE The instant messenger should be integrated into Solid Edge as a *sidebar* on the right side. The sidebar should be invisible when not needed and get unfolded with a single click.

PUSH NOTIFICATIONS On new messages of conversations the user does not currently view, a notification popup box should appear on the screen of the receiving user. The popup box should appear in the right corner of the screen and display the text of the new message.

CONVERSATION AND CONTACT LIST On the main interface of our instant messenger, the user should be able to see his open (meaning: active, not archived) conversations and a list of his contacts. By clicking on an open conversation, the conversation is displayed.

By clicking on one of his contacts, the current open conversation with this contact is showed. If there is no open conversation between the two, a new conversation is initiated and showed.

DETAIL VIEW FOR CONTACTS WITH SYSTEM AND OS INFO A user can view his counterpart's profile when clicking on the user's profile picture in a conversation. The profile should include information about used Solid Edge and Windows version.

ONLINE-STATE OF CONTACTS It should be possible to see when a contact is online and to see the time when a contact was online the last time. This should be possible for the contact list (online status for all contacts should be visible) and within a conversation (online status for conversation partner should be visible).

NAMED ARCHIVE FUNCTION It should be possible to archive a conversation and give it a title under which it can be found in an *Archived Conversations* list.

## 5.2 NON-FUNCTIONAL REQUIREMENTS

SECURITY A user should only have access to EmbeddedChat functionality if he can authenticate himself with proper login data. The password should not be stored in the database in plain text, but as an encrypted hash.

DATA STORAGE Conversation and contact data should not be stored on the user's local computer, but on a server. As a result, users can use EmbeddedChat independently of their local machine and do not have to worry about storage limitations.

IMPLEMENTATION CONSTRAINTS The core chat functionality shall be implemented as a web application with HTML5 web technologies, namely Angular.js (AngularJS) and Semantic UI (Semantic UI), and embedded into a Solid Edge sidebar. This offers an easier development and creates space for a broader use case because the web application could be integrated into every kind of program without much effort. It even could act as a standalone application. There exists the vision to combine all implemented social functionality at our chair into one powerful platform. Therefore, on the backend, Node.js (Node.js) as webserver and MongoDB (MongoDB) as database shall be used to maintain compatibility to Robert Gleixner's backend for SocialEdge Q&A (Gleixner, 2015).

PERFORMANCE CONSTRAINTS Messages should be delivered to the receiving user within a maximum time duration of 5 seconds.

# 6

# SOFTWARE DESIGN

Based on the requirements specified in section 5, the prototypical *EmbeddedChat* application was developed. In this chapter, we will describe its system architecture, underlying data model and the design of its three parts.

## 6.1 SYSTEM ARCHITECTURE

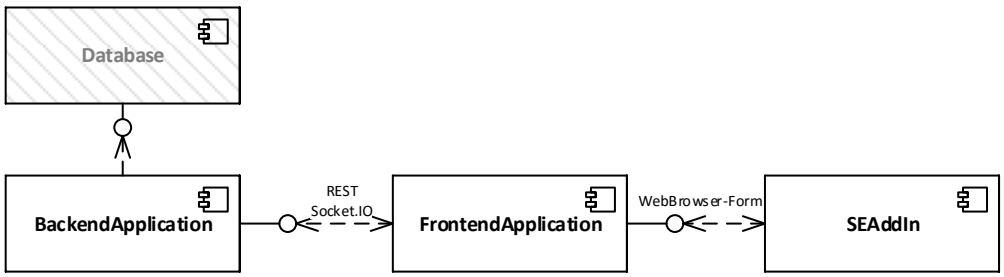



Figure 11: Architecture of EmbeddedChat System (Database not counted as a separate Part)

The EmbeddedChat system consists of three components and can be seen in figure 11. Users see and use the client application called *FrontendApplication*. It contains the whole user interface and program logic to connect with the central server *BackendApplication*. BackendApplication acts as a connector between clients and database. All data like past chats and user profiles are stored in the database and clients can access it by sending a correctly authenticated request to the server. Additionally, BackendApplication pushes new messages to connected clients. Despite being shown in the architecture figure, we do not count the database as a separate part because it is bounded to BackendApplication very tightly. Together, BackendApplication and FrontendApplication form a client-server architecture.

The component responsible for integration into Solid Edge is called *SEAddIn*. It adds an *EdgeBar*, the Solid Edge naming for "sidebar" to the user interface. Within this sidebar, FrontendApplication is shown.

## 6.2 DATA MODEL

This section describes the data model of EmbeddedChat - a graphical representation in UML is shown in figure 12. The data model is build out of four main entities: *User*, *Conversation*, *Message* and *Image*. Every data object of an entity has a unique *_id* which is also used for references between objects.

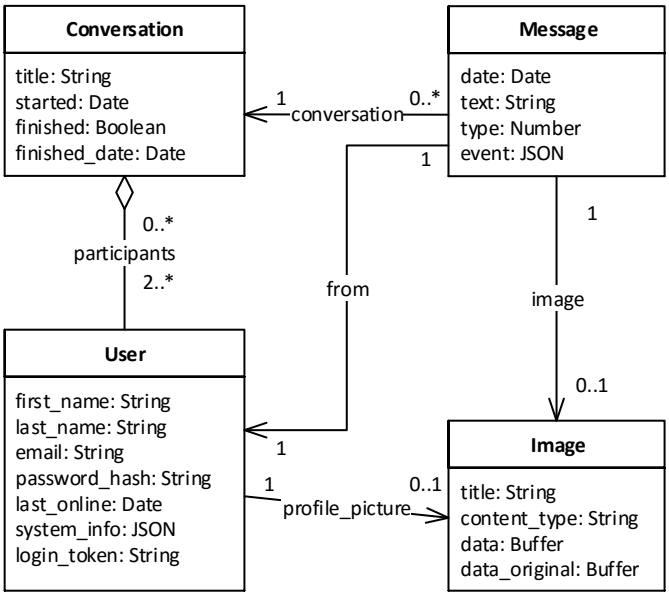



Figure 12: EmbeddedChat Data Model

The Conversation model includes meta information about a conversation, most importantly the references on the users participating in the conversation. A conversation can be active or finished/archived.

The User model contains login information as well as user information - name, last online state and system information. The system information is split up in fields *se_version* and *win_version*, Solid Edge and Windows version of the user.

A Message references to a conversation and can be one of three different types:

| **Number** | **Type** |
|---|---|
| 1 | Text message |
| 2 | Image message (a reference to an Image object) is stored |
| 3 | Message about a Solid Edge event (e.g. User started using the *Select*-Tool; User saved the document).<br>The event information is stored in JSON within two fields: The *event* type (name under which Solid Edge treats the event) and *payload*, plain additional event information of various kinds. |

Table 3: Different Types of Messages

## 6.3 FRONTENDAPPLICATION

### 6.3.1 *Technology*

FrontendApplication is a *single-page web application* implemented with modern HTML5 technologies. Single-page applications (SPAs) are "Web apps that load a single HTML page and dynamically update that page as the user interacts with the app" (Wasson, 2013). This means annoying page reloads are not needed. SPAs are characterized by the use of much client side code, mainly in JavaScript. In traditional web applications, the server delivers complete HTML web sites to the browser each time e.g. a button is clicked. In SPAs, after loading the initial page, only plain data is transfered furthermore; the page is updated accordingly via client side code. As many SPAs, we use the *JSON format* for data exchange.

We chose *Angular.js* (see AngularJS) as JavaScript SPA framework. Angular.js offers several concepts to facilitate application development and code structure. Angular.js applications are separated in independent modules. Besides the modules developed on one's own, developers can use modules published by other developers. Amongst others, we use module angular translate to localize our application both in English and German.

For user interface design, we chose *Semantic UI* (see Semantic UI). It includes several well-designed UI components which can be easily applied to HTML elements. Furthermore, it is responsive, meaning UI elements dynamically adjust to the available screen size. Because EmbeddedChat will be displayed in a dynamically resizable sidebar in Solid Edge, responsive design was very important. As reference width for the sidebar and testing width for our user interface we used a value of *360px*.

For some features, mockups were created to determine which concept would be the best. We either drew mockups on paper or used the software *Balsamiq Mockups 3* (Balsamiq) to create more realistic drawings on how the functionality would look like when implemented.

### 6.3.2 *States*

FrontendApplication is separated into 8 states with 4 additional nested states as can be seen in figure 13. The most important states are *start* and *chat* with their nested states. By using EmbeddedChat, a user navigates between the different states.

After logging in, the user enters the *start* state where he chooses between viewing a list of his active conversations and his contact list (default: *conversation-list*). From *conversation-list*, he can either go to an active conversation or view the list of archived conversations. From *contact-list*, he can create a *one to one conversation* or a *group conversation* (with *create-new-group-conversation* in between, to name the conversation and select participants). A user also can view and change his own profile (*view-profile*).

In *chat* state, the user can choose between viewing messages and images (nested state *chat*) or the Solid Edge event *timeline* (default: *chat*). The user can look at a detailed

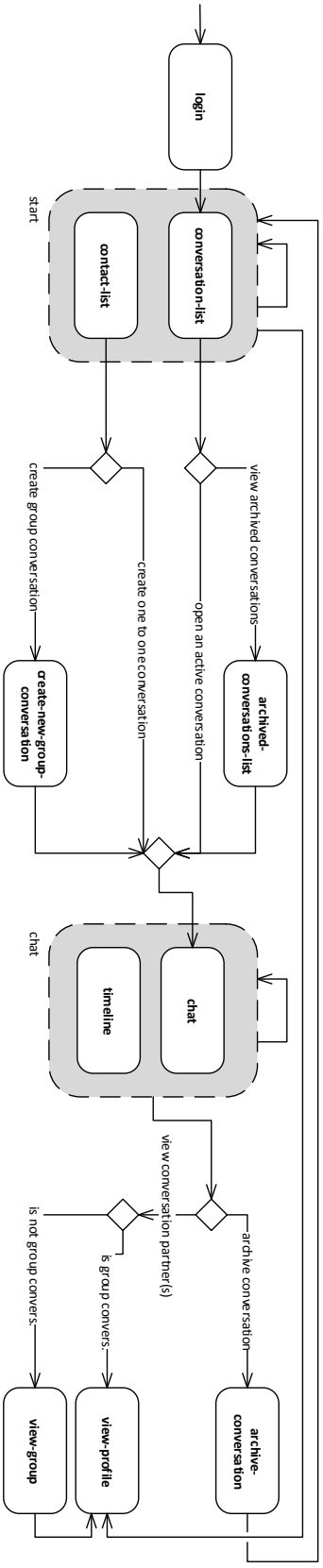



Figure 13: FrontendApplication Component Flowchart. Actions to go back were generally omitted from the Diagram to maintain Clarity

view of his conversation partners - either of the whole group on group conversations (*view-group*) or the single conversation partner (*view-profile*). Finally, a conversation can be archived and named by entering the *archive-conversation* state.

### 6.3.3 *User Interface*

In this section, we will show our user interface design, previous mockups and and detail on design decisions. Our user interface consists of several views - each view is coupled to a navigational state (for the states, see section 6.3.2). Some views were too simple and not interesting enough to show them in the main part of this thesis. You can find them in appendix 9.2).

MAIN INFLUENCES Our main influences for designing the user interface of EmbeddedChat were WhatsApp and Facebook Messenger, with its design used on messenger.com. They both provide a modern minimalistic, mobile-centric design which we found well-suited for our approach. Because the space available for EmbeddedChat (about 360px in width) resembles more to a mobile device, we used inspirations from mobile design a lot. WhatsApp and Messenger both have a large user base which proofs their designs as accepted by the users. We expected an increased ease of use and general well-being when users are reminded of messengers they use in their private life.

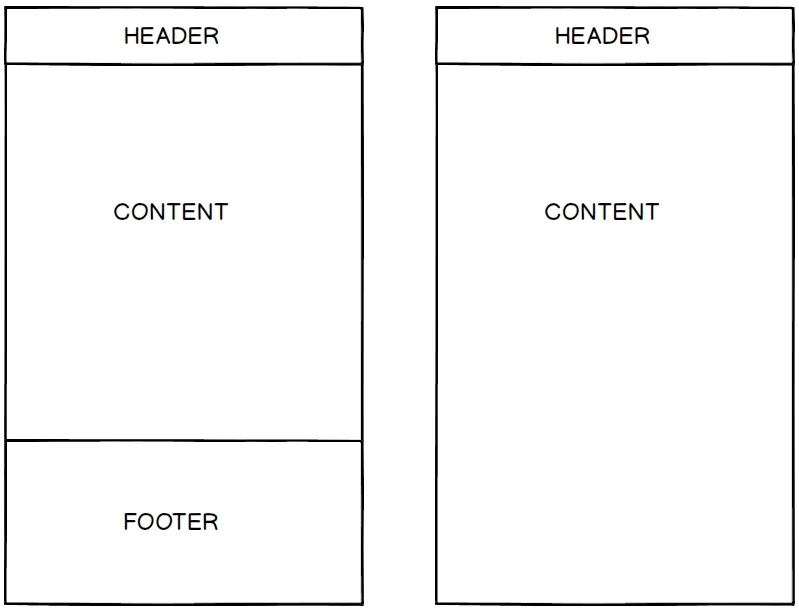



Figure 14: Concepts of a User Interface View

A view in EmbeddedChat is generally built up by three modular parts, see figure 14: Header, content and a footer. So, parts that appear on multiple views only have to be defined once. Changes only need to be applied to the single module and get displayed in all views which use it. The footer part is only used by state *chat* - in the other states, the footer is left empty and the content part fills up the remaining space.

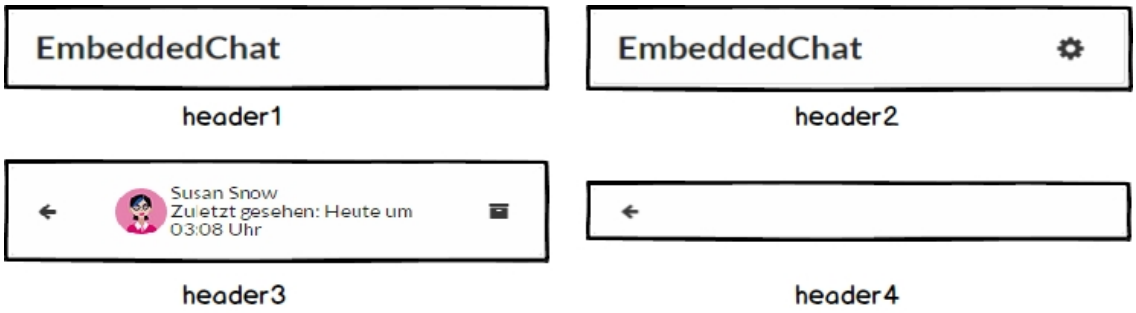



Figure 15: All four different Headers used in EmbeddedChat

We created four different headers (figure 15). Header 1 is only used for the login screen and does not contain any possibility to interact because the user should concentrate on authenticating oneself. Header 2 is shown when viewing one's contact or conversation list and has a settings button where the user can access his profile. While being in a conversation (state *chat*), header 3 is shown. A user can leave the conversation by clicking the back button, view information about his conversation partner in the center (a click leads to the user's profile) and archive the current conversation by clicking the button on the right. For all other states, the simple header 4 is displayed, which contains a back button to go back to the previous state.

#### 6.3.3.1 *Start Screen*

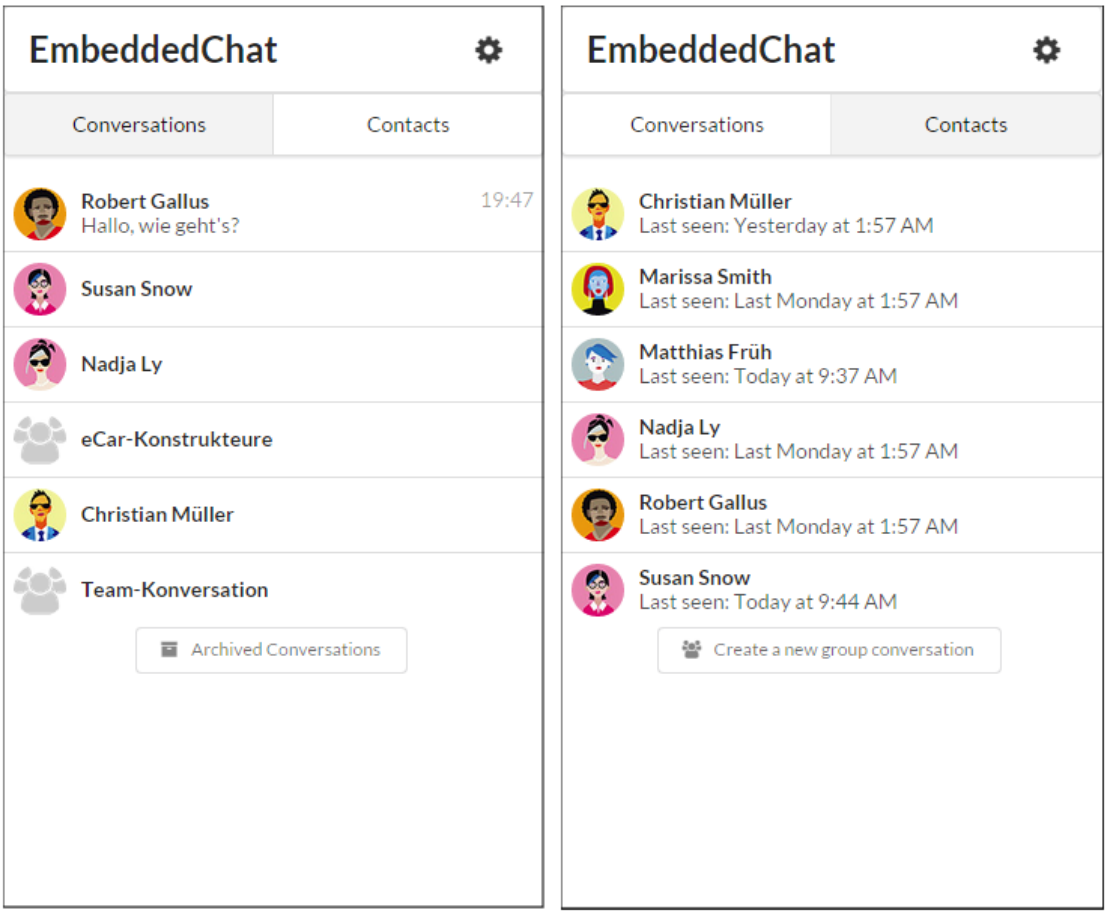



Figure 16: Left: List of open Conversations, Right: Contact List

When entering the start screen (figure 16, left), all open conversations are listed. For each conversation, profile picture of the other user and his name is shown. Below the user's name, the last message in this conversation is shown with send date on the right side. All conversations are ordered by the most recent last message. Under the list of last conversations, a button to view archived conversations is placed.

Via a tab switcher like in WhatsApp, users can switch to their contact list. An item in this list is similar to a conversation item. Instead of the last message, it is shown whether a user is online or was online time the last time. If it was not so long ago, indications like *5 minutes ago* or *Yesterday* are displayed - if more time has passed by, just the date. Under the contact list, a button to create a new group conversation is shown.

#### 6.3.3.2 *Chat View*

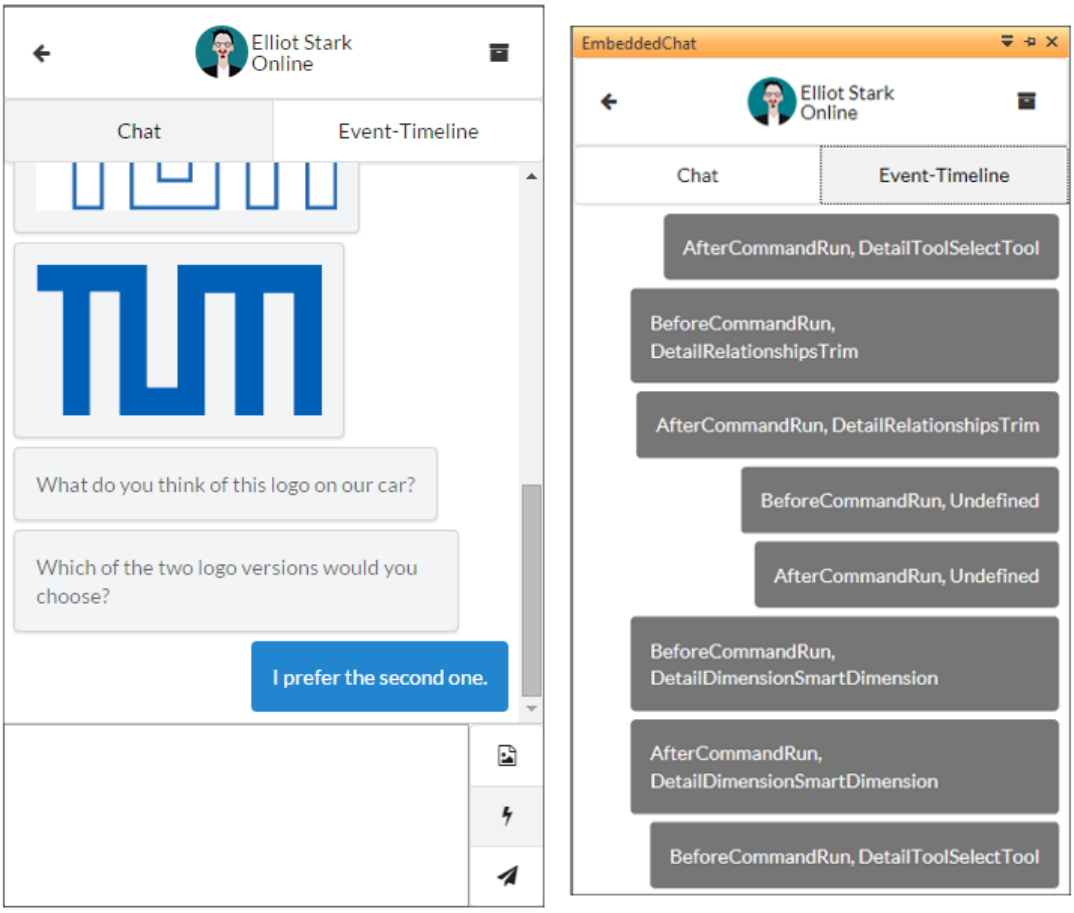



Figure 17: Chat View in a One-To-One Conversation. Left: Message View, Right: Event-Timeline View

In *chat* view, there exist two different tabs: One for chat messages and images and one for past events (*Event Timeline*). As it is in most of today's chats, newer messages are displayed at the bottom, left aligned messages signalize received messages, right aligned messages sent messages. To display the time a message was written, a user needs to hover the message. In the footer a text area with three buttons is displayed. The first button is related to sending images - by hovering the button, a menu appears where users can either click on a button to select an image from their hard drive or to directly send a screenshot. The images are, when sent, displayed as a normal message with the image scaled down to a maximum width of 200px. By clicking on the image, a large version is opened in a browser window.

The second button is an on-off switch for the *Event Timeline* functionality. By default, all commands used by one's own in Solid Edge are transmitted to EmbeddedChat and sent to all conversation partners. The toggle deactivates the transmission.

The last button is used to send a message with the text entered in the text area. Alternatively the shortcut *Shift+Enter* is available.

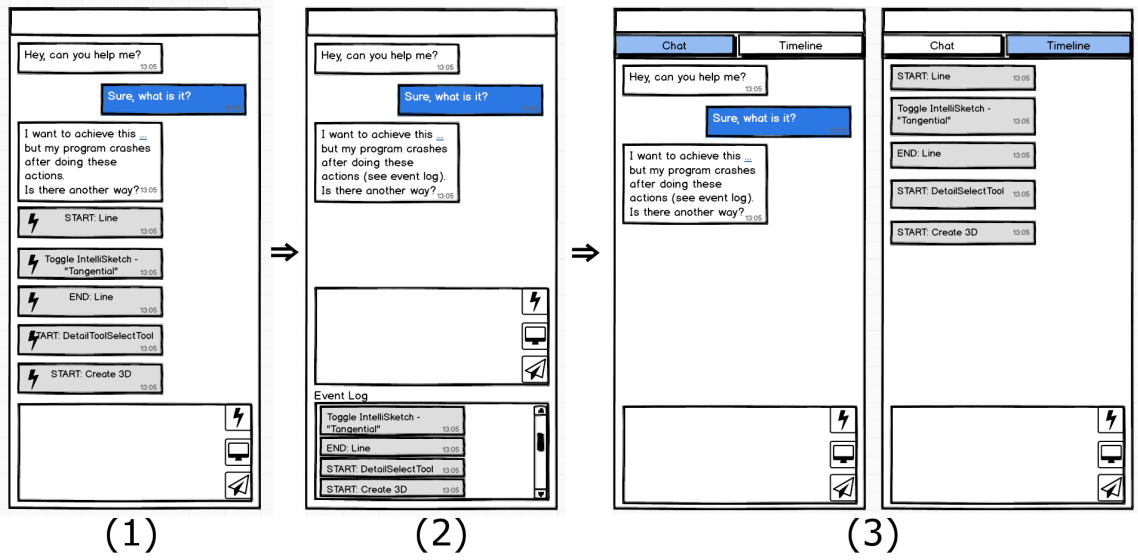



Figure 18: Evolution of Event Timeline Interface

One of the design choices where we had to create several mockups for was how to visualize the incoming command events when a participating user invokes a command. We developed an evolution of three concepts as can be seen in figure 18. At first, we though about treating events as normal messages with a slight different highlighting and display them together with text messages and images. This would resemble to current popular messengers where all types of content are combined in one area. But we discovered that events would probably pollute the conversation - if a conversation is open some minutes while one uses Solid Edge in a normal way, a lot of command events are generated, even if not important in this situation, and would make it difficult to follow the normal conversation. So we came up with an area below the input text area, where events should be displayed separately. Drawback with this solution is that this event log is taking too much space in times when information about command events is not needed. So we reused the *tab* pattern from the start screen. Events are displayed in a separate tab and therefore only visible when really needed.

#### 6.3.3.3 *Profile View*

In the profile view, a user can see basic information about other users, change his own information or log out from EmbeddedChat. To change information like name and e-mail, a user has to click on the respective text, enter the new information in the upcoming input field and click *Save*.

#### 6.3.3.4 *UI for group messages*

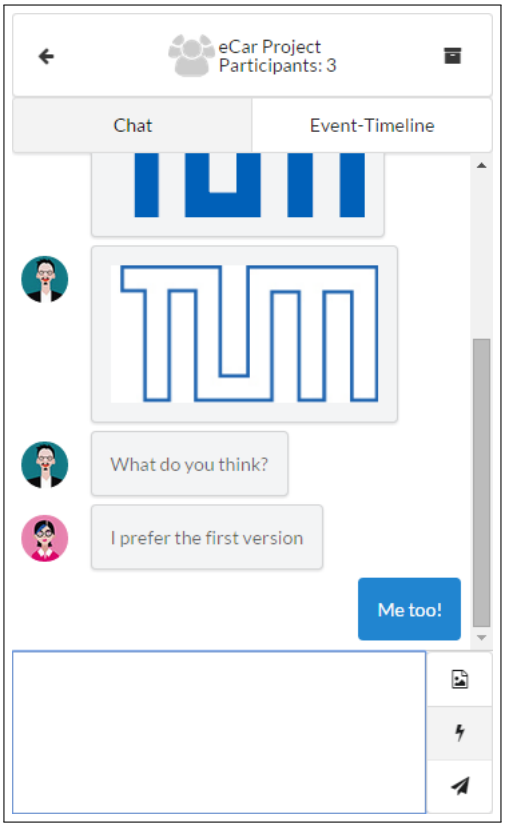



Figure 19: Group Chat. User Profile Pictures are displayed next to their Messages

Some UI changes and additions exist to handle group conversations. For creating a new group conversation, a separate view is needed. In this interface, the user needs to choose a conversation title and members of the group. He can only choose out of his contact list. From now on, where the name of the conversation partner was displayed so far, the conversation title will be. As profile picture, a special group icon will be displayed.

To see which member has written a message in a group conversation, the profile picture (which links to the user profile) is displayed next to the message (see figure 19.

## 6.4 BACKENDAPPLICATION

### 6.4.1 *Technology*

The BackendApplication is a web server application implemented in JavaScript, too, using *Node.js* (Node.js) and its framework *hapi* (hapi.js), which reinforces modular application logic. The use of JavaScript both on client and server side makes development easier.

### 6.4.2 *Communication with FrontendApplication*

BackendApplication's sole purpose is to react on requests from FrontendApplication clients and act appropriately. For communication with a FrontendApplication, two different concepts are used: With *RESTful* (REST: Representational State Transfer) requests, a classical "client request -> direct response from server" scheme is implemented. They are used by the client to ask for the list of one's open conversations, creating a new conversation, changing user information, getting an image etc. REST has some drawbacks in terms of real-time communication - there is no permanent connection between a client and server and the server can only respond to a client's request, but not initiate a request itself. So incoming messages could not get delivered to clients. The solution for having a permanent bidirectional connection between client and server and therefore be able to *push* messages to the client in real-time, are *WebSockets*. We use the JavaScript framework *Socket.IO* (Socket.IO) which simplifies the use of WebSockets and adds fall back mechanisms in case a browser does not support WebSockets.

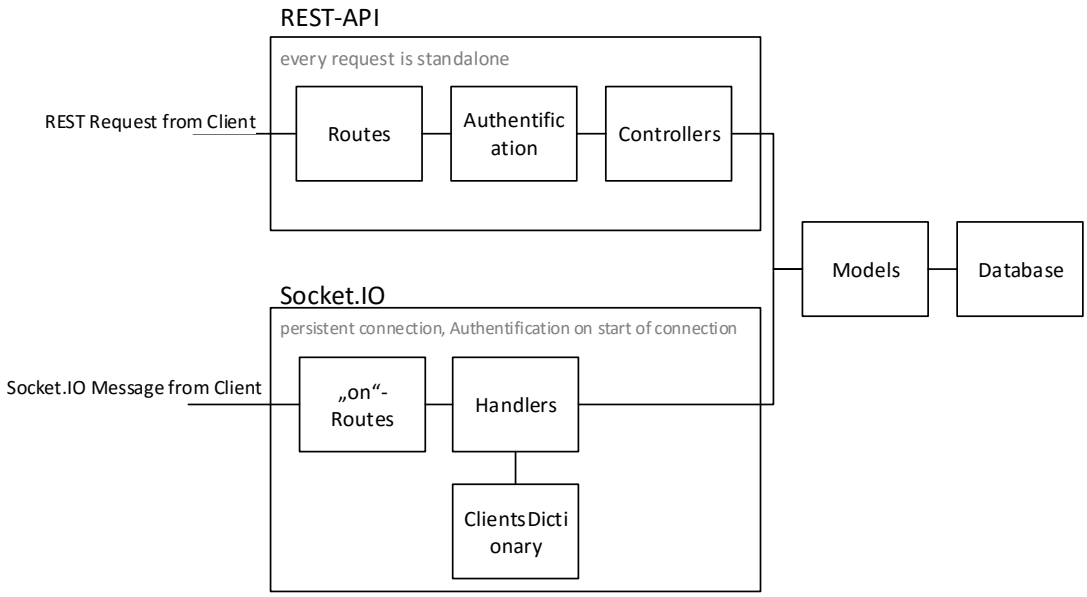



Figure 20: Visualization of Different Types of Communication between Client and Server

#### 6.4.2.1 *REST API*

The basic data type in REST is a *resource*, a data type like *Message* or *Conversation* defined in section 6.2. A client can access and modify a resource by calling a URI (Uniform Resource Identifier) with a specific HTTP method as verb: *GET, POST, DELETE or UPDATE*. He can add data to his request which, for example, provides the new data the updated resource should have. The detailed API (Application Programming Interface) documentation is shown in the appendix 9.3.1.

The typical REST request process is depicted in figure 20. When a request is received by the server, it is checked to which URI the request was directed and with a table of *routes*, an appropriate *controller* is chosen to process the request. Before the controller's code can process the request, it has to be checked if the requesting client is logged in and therefore authorized to make the request.

AUTHENTICATION REST communication is stateless. This means, the server treats every request as independent and does not keep track of a clients state. Each request from client to server must contain all the information necessary to understand the request. Session state is therefore kept entirely on the client (Fielding and Taylor, 2000). Our API differs from REST's standard constraints in some parts: To keep an user authenticated and skip the need to add full authentication information to every request, we generate a token after a successful login and save it on the server (loose implementation of Hardt and Jones). Now, the client can proof its authorization just by adding the token to every request.

DATABASE Almost every request is related to a database operation. We are using *MongoDB* (MongoDB) with the Node.js object modeling framework *Mongoose* (Mongoose). Every data type defined in 6.2 is mapped in Mongoose - it adds methods to these data types to create a new object of a data type or change attributes for a specific object. Overall, we do not need to bother directly with MongoDB programmatically.

When the needed database action is finished, the server answers with a response. This could be objects the client requested, just an empty response stating a succeeded request or an error if something went wrong processing the request.

#### 6.4.2.2 *Socket.IO - Real-Time Communication*

When a client connects to the server via Socket.IO, authentication is done once at the start of connection. The client is marked as online by being added to an internal *Clients-Dictionary*. Both client and server can now emit events with, for example, data of a new message, to the common socket. The server is connected to all online clients and can emit events not only to a specific client or all clients, but also to clients who take part in a specific conversation. Socket.IO uses a *room concept* to achieve this - each *conversation_id* identifies one room, clients can be added to a room so they get notified when a new message arrives.

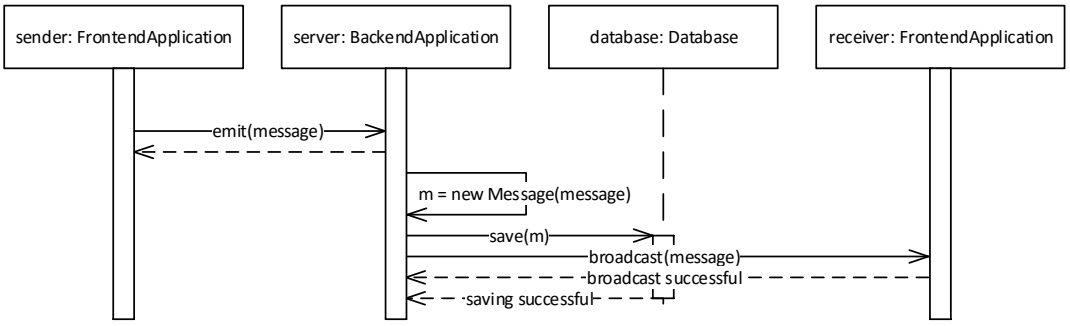



Figure 21: Sending and Receiving of a Message with Socket.IO

Figure 21 shows the procedure for a new message being posted to a conversation. Initial position is clients *sender* and *receiver* being connected to server *server*. Now, *sender* posts a new message in a conversation both clients are in. He emits the message to the server, where a new *Message* instance gets created. This instance is saved to the database - without waiting for the save to be complete, the server broadcasts it to all connected clients within the same conversation, except the sender.

Like with routes in REST requests, we defined different types of events which can be emitted to a socket and generate different server behavior. A detailed list with descriptions can be found in appendix 9.3.2.

## 6.5 SOLID EDGE ADDIN

### 6.5.1 *Technology*

The FrontendApplication web application needs to be integrated into the Solid Edge environment as a sidebar. For achieving that, Solid Edge offers a *COM* (Component Object model) based *API* (Newell, 11.10.2014; Siemens). The C# framework *SolidEdge.Community* (see GitHub) offers much of the needed code and was used for add-in development.

Because our sidebar only needs to contain a component which can display a website, we use Windows Forms' *WebBrowser* (see Microsoft). This form uses the newest Internet Explorer installed for displaying the website. An unfortunate problem we faced while developing was that websites were rendered in a way Internet Explorer 6 or 7 would have done it, even if Internet Explorer 11 was installed, probably for compatibility reasons with old websites. This led to several displaying problems which we could solve by setting a specific registry entry (*FEATURE_BROWSER_EMULATION*). We describe this more in detail in the add-in README file.

We tried, but did not succeed in embedding solutions like CefSharp and GeckoFX in our sidebar to have a different rendering engine and to be able to use modern features like WebRTC with webcam chat and screensharing. In the end, it should be possible somehow and may be a good feature addition for the future.

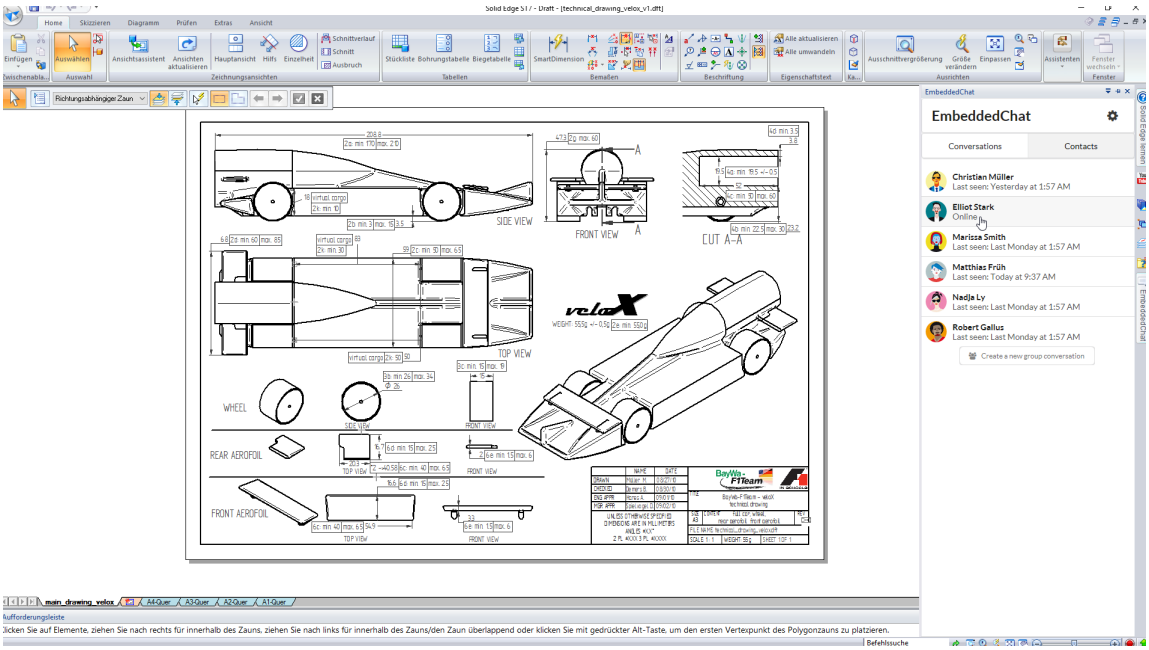



Figure 22: Screenshot of Solid Edge with integrated EmbeddedChat

### 6.5.2 *Communication with FrontendApplication*

Although most of the time the add-in sidebar is just a container for FrontendApplication, some features require communication between those two components. We found a way to invoke methods in both ways - FrontendApplication can invoke specific C# add-in methods and receive a result, the add-in can invoke JavaScript methods.

SCREENSHOT If a user wants to take a screenshot, a sole web page has too limited system access to accomplish this task. Therefore it calls a C# add-in method which captures the main screen content and returns the image back to the web application encoded in base64, which means as a plain text string (for more information, consider Wikipedia (2015a)). The base64 string then gets uploaded to the server and converted to a normal picture.

NOTIFICATIONS When a new message comes in, the user should get notified without having to have the sidebar open all time. So again, a C# method with the notification content as parameter gets called by FrontendApplication. The C# method creates a new popup box which is displayed outside the web page, in the right corner of the normal Windows screen.

EVENT TIMELINE For the command event transmission (Event Timeline), a C# method needs to call a JavaScript function with the event name and information as parameters. Within the add-in, we add methods which are called before and after every application event. An object containing the type of the application event and a payload with informa-

tion like the command name is generated[1] and transmitted to the web application as a JSON string. The web application can then send the event to the server and then display it appropriately in the *Event Timeline*.

### 6.5.3 *Deployment and Installation Instructions*

To use EmbeddedChat in a live environment, all parts need to be deployed. In a test deployment, we uploaded the FrontendApplication to a simple HTML-serving webspace. Because this part only consists of HTML, CSS and JavaScript files, it could be saved on a local machine without the use of special tools, too. For BackendApplication, a web-hosting provider which enables use of Node.js applications and MongoDB database is needed. We used Heroku which allows us a simple codebase upload via `git push`. In production, FrontendApplication and BackendApplication could be run on the same server with FrontendApplication being just served as static files. The AddIn part needs to be registered to the Windows Registry to show off as a Solid Edge add-in. A more detailed description on how to get EmbeddedChat running on an own environment can be found in the README files delivered with the EmbeddedChat code repository.

1 A useful tool for discovering the powerfulness of the Solid Edge API to later implement it into our add-in was Solid Edge Spy (Newell, 2014)

# 7 EVALUATION

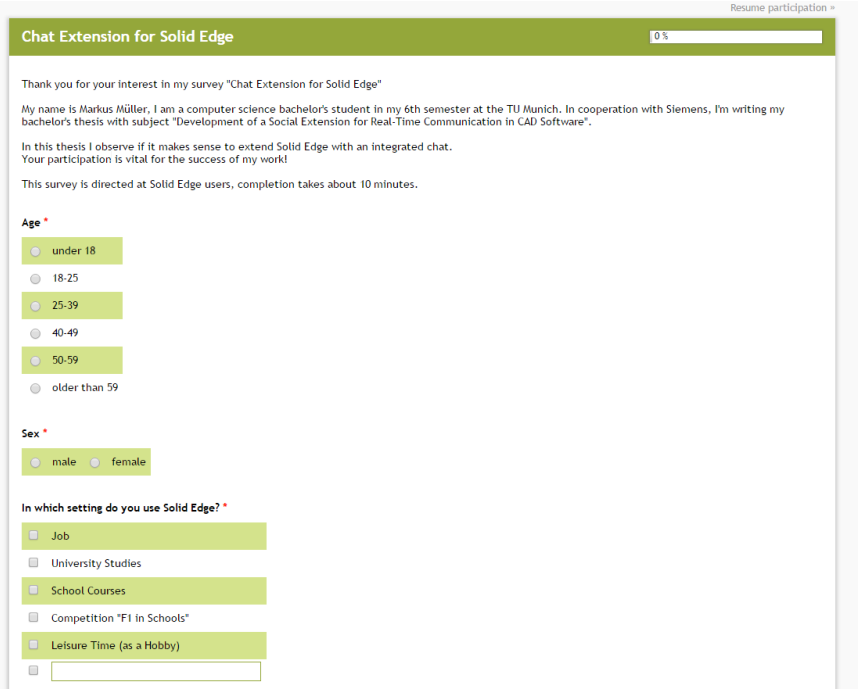



Figure 23: Screenshot of online evaluation

To answer the last research question *"Do CAD designers find instant messaging a useful addition to their job?"*, an online-survey was conducted (screenshot 23). Section 7.1 introduces the general evaluation process and structure, section 7.2 presents the results. In section 7.2.4, we will discuss findings and effects of the results, but also will go into detail about probable evaluation's limitations.

You can find the whole evaluation questionnaire and detailed results in appendix 9.4.

## 7.1 EVALUATION DESIGN

### 7.1.1 *Hypotheses*

Coming from our research question *"Do CAD designers find instant messaging a useful addition to their job?"*, we introduced three hypotheses which had to be checked in the online-survey:

- H1: Solid Edge users often use chat messengers in their private life, but do not use any for work communication.

- H2: Solid Edge users find chatting with their (Solid-Edge-using) contacts within the program useful.
- H3: Solid Edge users see a need for chat functionality within Solid Edge at their workplace

### 7.1.2 *Structure*

The evaluation is separated into four parts. In the first part, we ask users general questions to take their later answers into account correctly: Users answer questions about themselves (age, gender) and how much and what for they use Solid Edge.

Each of the next three parts is matched to one of our hypotheses presented in subsection 7.1.1. To verify "H1: Solid Edge users often use chat messengers in their private life, but don't use any for work communication.", users shall first describe their number of contacts related to their Solid Edge work and specify how much of their contacts use Solid Edge regularly, too. They should mark which means of communication they use how much to communicate at work, personal meetings and phone calls being some the choices. Secondly, they indicate how much they use chat messenger in private life.

Then, a YouTube video which presents our *EmbeddedChat* is shown. It gives a three minute long presentation on main features and two use cases:

| | **Action** | **Feature presented** |
|---|---|---|
| 1 | User is within Solid Edge, opens EmbeddedChat EdgeBar and logs in | Integration into Solid Edge |
| 2 | User switches from conversation list to his contact list | View online status of contacts |
| 3 | User starts a new chat and sends a screenshot to the contact | Direct screenshot sending |
| 4 | User tries to export his document as PDF, asks other user | |
| 5 | By examining the event timeline, the other can tell what our user did wrong | Event Timeline |
| 6 | Users archives the conversation | Conversation archivation with title |
| 7 | User gets a notification | Notifications on new messages |
| 8 | User enters the group conversation in which he got the message | Group Conversations |
| 9 | Other user asks group about opinion and sends two different images to choose from | Image Sending |

Table 4: Demonstration Video Storyboard

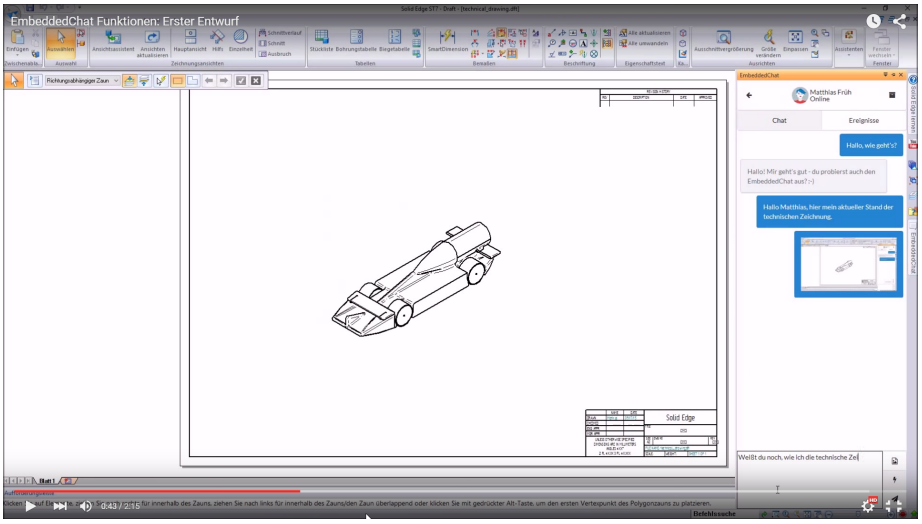



Figure 24: Frame of Demonstration Video

After viewing the video, to verify "H2: Solid Edge users find chatting with their (Solid-Edge-using) contacts within the program useful.", questions from Davis (1989) in the context of his *Technology Acceptance Model* are used to measure the perceived usefulness. Then users are asked how important each of the shown features is to them.

To check our last hypothesis, "H3: Solid Edge users see a need for chat functionality within Solid Edge at their workplace", we ask users to estimate the ease of learning the usage of EmbeddedChat, ease of persuading their contacts to use it and their estimation whether EmbeddedChat would get established as a means of communication successfully.

### 7.1.3 *Target Groups*

On the one hand, we sent links to the survey to our company contacts we already had from the qualitative interview phase. On the other hand, we called for participation in user groups (Facebook and LinkedIn) and discussion boards (CAD.de and the official Siemens Solid Edge community discussion board). Because our audience would be anglophone as well as German-speaking, we created the survey for both languages and merged the results.

## 7.2 EVALUATION RESULTS

### 7.2.1 *Reactions*

In the two discussion boards we campaigned for survey participation, the German discussion on CAD.de (Müller, 2015) generated over 550 views with 12 replies from 6 members (us included), the English Siemens PLM community discussion generated over 330 views with 5 replies from 4 members (us included). The main opinion on CAD.de was

that an integrated instant messenger like EmbeddedChat is not needed. There would exist enough good standalone instant messenger applications if one wants to use chat functionality. At the PLM community discussion, feedback was more positive and two members named EmbeddedChat a good tool.

The discussion on CAD.de became quite heated as one member accused the survey to be suggestive. The survey would "postulate without saying" that a chat is wanted and makes sense. This is interesting because most of the questions the person probably meant were directly taken from Davis (1989) to measure the perceived usefulness. Furthermore, the commenter made a connection from EmbeddedChat to *"cost-free" programs* whose only function is to collect data from users.

### 7.2.2 *Participants*

91 people participated in the online survey, of which 35.1% discontinued before reaching the end. 68% of participations where done in German, 32% in English. 25 and 13 persons added their own comments within the two free-text questions. 80% (73 times) of participants stated that they use Solid Edge in a work-related setting, 14.3% (13) name F1 in Schools as their setting of use. Multiple answers were possible. Answers on how many hours per week Solid Edge is used ranged from 0.5 to 60 with most of the people working 20 to 40 hours per week with Solid Edge.

Participants selected *Construction of new parts* (on average, 40.3% of their time), *Giving support to other Solid Edge users* (on average, 33.4% of their time) and *Modifications on existing parts* (on average, 29.2% of their time) as their primary categories of usage of Solid Edge. Usage *Giving support to other Solid Edge users* probably gained that high number of votes because some full-time support employees participated. It was possible to select his usage categories in a way to more than 100% time. We estimate that users did this to signalize multiple usage categories occur at the same time sometimes.

### 7.2.3 *Hypothesis Verifications*

| | several times a day (1) | | daily (2) | | every 2-3 days (3) | | weekly (4) | | every 2-3 weeks (5) | | monthly (6) | | less frequently (7) | | never (8) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Ø | ± |
| Meetings in person | 22x | 31,43 | 17x | 24,29 | 4x | 5,71 | 7x | 10,00 | 4x | 5,71 | 4x | 5,71 | 8x | 11,43 | 4x | 5,71 | 3,26 | 2,37 |
| E-Mail | 21x | 30,00 | 18x | 25,71 | 5x | 7,14 | 8x | 11,43 | 1x | 1,43 | 6x | 8,57 | 4x | 5,71 | 7x | 10,00 | 3,27 | 2,41 |
| Phone | 17x | 24,29 | 17x | 24,29 | 9x | 12,86 | 5x | 7,14 | 3x | 4,29 | 9x | 12,86 | 5x | 7,14 | 5x | 7,14 | 3,46 | 2,33 |
| Facebook or a Social Net… | 3x | 4,29 | 7x | 10,00 | 2x | 2,86 | 6x | 8,57 | - | - | 2x | 2,86 | 4x | 5,71 | 46x | 65,71 | 6,50 | 2,41 |
| Instant Messenger | 6x | 8,57 | 12x | 17,14 | 1x | 1,43 | 3x | 4,29 | 2x | 2,86 | 2x | 2,86 | 1x | 1,43 | 43x | 61,43 | 5,97 | 2,79 |

Figure 25: Answers to question *Which means of communication are then used? How often for each means?*

H1: SOLID EDGE USERS OFTEN USE CHAT MESSENGERS IN THEIR PRIVATE LIFE, BUT DON'T USE ANY FOR WORK COMMUNICATION. When asked on how often Solid Edge users use certain means of communication, meetings in person, mail and phone were mentioned the most by far (see figure 25. *Instant Messengers* were, from our predefined choices, the second-least often used means of communication, only *Facebook or a Social Network* being used less often. *Instant Messaging* got a score of 5.97 by average with 1 being *several times a day* and 8 *never*. So the second part of H1 could be verified.

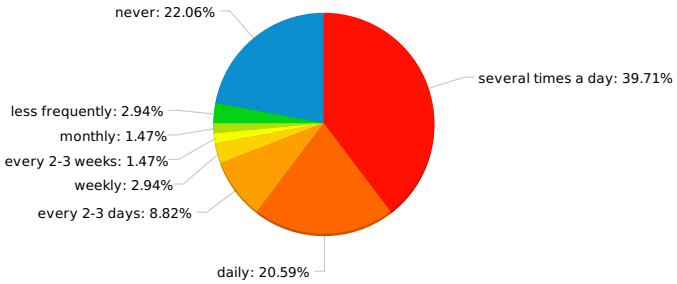



Figure 26: Visualization of answers to question *How often do you use instant (chat) messengers in your private life?*

As can be seen in figure 26, most of the participants use chat messengers several times a day for private matters - average is 3.32, if number mapping is the same as before. Overall, we could verify H1 completely. Solid Edge users mostly do not use instant messengers for work, although they certainly use them in their private life.

H2: SOLID EDGE USERS FIND CHATTING WITH THEIR (SOLID-EDGE-USING) CONTACTS WITHIN THE PROGRAM USEFUL. After watching the demo video of EmbeddedChat, users should evaluate its usefulness. All questions regarding the perceived usefulness resulted approximately in *Neither* (4 on a scale of 1 being extremely agreeing and 7 extremely disagreeing), with a slight tendency to disagreeing, meaning not useful. It is to add that noticeable more users *extremely disagreed* with statements which expressed the usefulness of EmbeddedChat than *extremely agreed*. On elaborating there choice for question *I would find EmbeddedChat useful in my job.*, often seen counter-arguments were "People I would need to communicate with would not use Solid Edge" and "The chat would mainly be a distraction.". One exemplary comment summarized this:

> *Sorry but the idea of using "Embedded Chat" is just another distraction from people doing their jobs. Could I see having the occasion to use this... maybe. But the chances that the people I would need to go into this kind of detail with, would NOT be using solid Edge or other CAD programs. So what's the point?*

Sometimes people reported they are already using an instant messenger or have already tested the introduction of instant messengers at their workplace. Others suggest features for more extended collaboration and support like *screen sharing* and *remote control*.

When asked on how important each of the shown features is to them, all features ranked at an average between 3 and 5 (1: very important, 8: absolutely unimportant). In this

small interval, screenshot sending is seen as most important feature (3.49), integration into Solid Edge is seen as least important (4.49), closely followed by "See recently used commands of your contacts".

In the end, we do not have a clear result to make statements over H2 but see a tendency to disapprove it.

H3: SOLID EDGE USERS SEE A NEED FOR CHAT FUNCTIONALITY WITHIN SOLID EDGE AT THEIR WORKPLACE When asked if an instant messenger like EmbeddedChat is needed at their specific workplace, results accord with the results for the previous hypothesis and show that an introduction is more likely not favored. Users generally agree that learning EmbeddedChat use would be easy for them. Most of them would have a clear vision for which purpose they can use EmbeddedChat. Again, opinions against an introduction are generally less in numbers, but more extreme.

As for H2, we more see a tendency to disapprove it than to prove it right.

### 7.2.4 *Comments and Discussion*

#### 7.2.4.1 *Comments*

Some participants added suggestions in the final comments section to improve EmbeddedChat or take it to another direction. An interesting suggestion is to develop a mixture of OneNote and Skype, integrated into Solid Edge. Another suggestion is to integrate EmbeddedChat into Insight, the SharePoint-based data management for Solid Edge. A benefit would be that users could get informed on important processes by the system - we think of something like the application integrations seen in Slack (see paragraph 3.2.2.2). This could be a way to improve context information and out of it create more value for using an instant messenger like EmbeddedChat. Chat functionality alone seems not to be a reason for users to want an integrated tool.

#### 7.2.4.2 *Discussion*

SOLID EDGE USERS ARE NOT UNITED WHETHER INSTANT MESSAGING IN GENERAL IS A USEFUL ADDITION TO THEIR JOB (DISTRACTION) Even if the majority of interviewees uses instant messaging regularly and therefore sees value in that, this can not be said for professional use. There exists great discordance about whether instant messaging is really useful or just distracting.

INTEGRATION IS NOT A DEAL-BREAKER, FAST ACCESS ON CERTAIN COLLABORATIVE FEATURES IS As we have seen at the answers on which feature people find most useful at EmbeddedChat, direct integration is not really important for them. Features for more a powerful communication and collaboration are more important.

EXTREME DISAPPROVAL AGAINST INSTANT MESSAGING IS MORE FREQUENT THAN EXTREME APPROVAL In quite a lot of questions we saw that extreme disapproval of instant messaging occurred more often than extreme approval. Even though the number of extreme disapprovals is lower than the number of relatively agreeing votes, we would rate those relatively agreeing votes less important than their mere number might suggest. It is more convenient for people to agree or check a neutral answer without a thought than to question the proposed concept in whole.

# 8

# CONCLUSION

## 8.1 SUMMARY

In summary, this thesis could largely answer the research questions formulated in the introduction. After giving an insight about current instant messengers, both for general and business-specific use and previous research on work-chats, we conduced several qualitative interviews to investigate how CAD designers communicate in their daily work. Then, the integrated instant messenger *EmbeddedChat* was developed with a focus on the *context* of Solid Edge. To evaluate positions on whether Solid Edge constructors find instant messaging a useful addition, with EmbeddedChat as prototypic example, an online survey was conducted. The results show that the concept of an instant messenger for work is still a difficult and heatedly debated topic with no definite answer. Like we have seen it in Herbsleb et al. (2002), there are several concerns about the benefit of instant messaging at work.

In retrospective, some parts of this work should have been done different. Research questions should have been defined more carefully to be able to answer them in a stronger way. A more specific use-case, with concrete evidence from real users, should have been designed prior to implementing EmbeddedChat. This would have made conceptual design and requirements engineering easier - in the end, we ended up with a quite generic instant messenger whose use compared to existing stand-alone messengers is debatable.

Originally, at the start of the development on EmbeddedChat, a more specific concept with focus on "problem-solving" was planned (4.1), but later terminated because of unsolved questions and being probably to strict in its problem-solving process. Our qualitative interviews in the beginning should have been designed to contribute more to such a specific use-case. Maybe because we had to build up the entire set of companies we worked with on our own and first had to get to know them and their work better, we left research for a specific use-case unattended.

At last, a real-world test would have been a better way to verify usability and additional benefit for users. An online survey can be useful because of the sheer numbers of participants, but long-term test feedback like in a real-world test is more well-founded.

## 8.2 FUTURE WORK

There are many possibilities where further work could be set - On the one hand, the EmbeddedChat could be still get enhanced as a generic instant messenger, but step by step with more features to facilitate communication. An example would be the addition

of screen sharing[1]. On the other hand, a specific process could be designed in cooperation with one reference enterprise to suit their needs the best. Then EmbeddedChat could be adapted according to this process. Conduction of a real-world use test would be useful no matter which option was chosen. A third possibility would be the integration of EmbeddedChat into more different programs and/or the introduction of a standalone chat client, all connected to the same backend. In the evaluation, we often heard the concern that all chat data would only be accessible through Solid Edge when using EmbeddedChat. Because of its implementation as a web application, this is not the case, interesting use cases might evolve.

Regarding the technical side of EmbeddedChat, the security and stability aspects need to get revised. Up to now, REST API requests do not check if the logged-in user has the correct rights to fulfill an arbitrary API request. Users so could easily change and read conversations of other users if they are logged in and create custom API requests manually.

In the context of our online survey, some companies expressed interest on an instant messenger like EmbeddedChat. Contacts made within the work on this thesis could be used to determine such a reference company and conduct a more extended evaluation.

1 Which would need a different browser engine than Windows Forms WebBrowser, either CefSharp or GeckoFX.

# 9

# APPENDIX

## 9.1 RESULTS FROM QUALITATIVE INTERVIEWS

Interview Results are paraphrased from the notes we took during the interviews. The interviews are ordered by time they were conduced.

### 9.1.1 *F1 in Schools Interviews*

All interviews occurred at the German Championship on May 8$^{th}$, 2015, at the Hockenheimring.

#### 9.1.1.1 *Team Vast Velocity, two constructors (later Championship Winner)*

*How did you acquire your Solid Edge knowledge?*

- We had a predecessor team
- The constructor of this team personally introduced us on how to work with the program
- We could ask the other team anytime we want

*How did you communicate when issues occurred?*

- Most of the time we used *WhatsApp*. Sometimes we really had to write very long messages to describe and solve problems. In doing so we took photos from the screen with our smartphones
- We then also met in personal

*What do you think of an integrated instant messenger for Solid Edge? There, you could send images and screenshots directly and see the last commands you and your chat partner used.*

- This would be a good thing, especially the *last used tools of chat partner* feature
- We would suggest integrated voice chat like Skype - we then could get the problem messages across easier

*Did you use standard Solid Edge help points like special discussion boards?*

- No, we *never* used discussion boards
- Sometimes, we took a look in the Solid Edge documentation. But the documentation is not sufficient in some parts.
- The error protocol Solid Edge sometimes provides to help was not very meaningful and we could not gather much it

#### 9.1.1.2 *Junior Team Millenium Tech, one constructor*

*How did you acquire your Solid Edge knowledge?*

- I had contact to a previous team. The constructor had already participated quite a lot of times.
- Especially in the beginning we often constructed together and met in person
- My teachers in school rather helped with the competition in general than with Solid Edge specific matters
- There was a Siemens training course at my school, but I was ill that day

*How did you communicate when issues occurred?*

- On problems, we had phone and Skype calls most of the time.
- When using Skype, we extensively used the screen sharing feature. My contact then showed me how to achieve what I wanted.

*What were common problems for you?*

- My most common problem was that I already tried a lot at a CAD model to achieve a certain result, but then forgot what I already tried. This resulted in long explanations and sometimes trying out the same things multiple times.
- I often had errors when rendering with KeyShot. In the end we used Cinema 4D for our renderings.

*Did you use standard Solid Edge help help points like special discussion boards?*

- I did not use any discussion board, but I think my helper probably used one to help me.

*What do you think of an integrated instant messenger for Solid Edge? There, you could send images and screenshots directly and see the last commands you and your chat partner used.*

- Would be cool and useful, especially features for seeing the commands already tried and previously used.

#### 9.1.1.3 *Team pursue, one constructor*

*How did you acquire your Solid Edge knowledge?*

- I am already working with Solid Edge for four years (since version ST2)
- In the beginning I had of course help from other persons, but I can't remember it exactly
- Now, I figure out most of the knowledge on my own. I use the internet a lot, there exist quite good tutorials.
- I have a contact to a professional Solid Edge user, but I have not talked with him a lot

*What do you think of an integrated instant messenger for Solid Edge? There, you could send images and screenshots directly and see the last commands you and your chat partner used.*

- I would have security concerns - the other one could maybe get the model file or get control over my mouse cursor

- The chat window should by all means not be visible the whole time, but only when it is really needed.
- Features a chat like this needs: Image and screenshot sending.
- The feature *protocol of the last action* would be also useful for oneself

*Do you have other recommendations which would help people with Solid Edge?*

- More videos to more advanced features would be useful.

#### 9.1.1.4 *Team Gasoline Racing, two constructors*

*How did you acquire your Solid Edge knowledge?*

- We do not use Solid Edge but Autodesk Inventor because we received help with this program from a F1 in Schools team in the same city
- We were also not able to use Solid Edge with the tutorials provided and had no real contact person for Solid Edge. The internet discussion boards were not sufficient and our internet searches were not fruitful

*How did you communicate when issues occurred?*

- We often met in person because then we could help each other the best.

*What do you think of an integrated instant messenger for Solid Edge? There, you could send images and screenshots directly and see the last commands you and your chat partner used.*

- For problem solution, the visual component is by all means very important to see what the other one is doing right now
- The *last used commands* feature would be useful, too
- We would appreciate a remote control feature because we often used the Skype screen sharing service.

### 9.1.2 *Company Interviews*

#### 9.1.2.1 *Small Company in the sensor system sector, 15 employees*

Interviewed one employee who works with Solid Edge

*What for do you use Solid Edge?*

- I am using Solid Edge to construct islands of machine tools and sensors
- I am using Solid Edge 3-4 hours a day.

*How do you solve issues connected to your work with Solid Edge?*

- When issues occur, we are on on our own because we do not have a maintenance agreement with a support provider.
- I have 2-3 co-workers at the same site who can help me
- So I use the internet a lot and am often reading discussions on discussion boards. I limit myself to 30 minutes when searching in the internet.

- Most of the time, not a specific command is the problem but the order of commands to achieve a certain result.

*How do you document such solutions for difficulties and problems?*

- We create PDF documents for our colleagues

*What do you think of an integrated instant messenger for Solid Edge? There, you could send images and screenshots directly and see the last commands you and your chat partner used.*

- This idea would make sense.
- A problem I see is finding contacts to communicate with. Outside of my company I do not know anyone who uses Solid Edge - so my contact list would be quite empty and I can not ask people for help.

#### 9.1.2.2 *Medium-sized company in the attachments for loader cranes and excavator sector with about 165 employees and 10-15 CAD employees working at the site we visited*

FIRST INTERVIEW *How do you use Solid Edge?*

- Currently we are using Solid Edge in version ST5, soon ST7
- I am working with Solid Edge about 5h a day and have already about 15 years experience in the CAD sector.

*What are typical issues with Solid Edge?*

- My main issues with Solid Edge are in the assembly functionality, relations between parts get destroyed sometimes
- Our hardware is a bit outdated, many problems maybe occur because of that.
- We only get trainings every 2-3 new versions, so we may not know about newer Solid Edge workflows

*How do you solve issues connected to your work with Solid Edge?*

- As colleagues we help each other. If a colleague who can help us is in the same room, we just pass by. Otherwise we call him and then send him a mail with screenshots.
- We built up an own intern shared archive for problem solutions containing Word documents with screenshots.
- If a problem can not be solved in our team or with our solution archive or we can not find a better way, we either do it as before or call or mail our support. This works pretty good. Normally after one or two days, after some files are sent back and forth, we can solve the problem together.
- If the problem can not be solved because it is a bug, we inform Siemens. But this takes a long time and we do not wait on the bug being resolved.
- The help in Solid Edge is pretty good, although I have not used it in a while.
- I do not search for a solution in the internet that much.

*What do you think of an integrated instant messenger for Solid Edge? Who would be your contacts in an instant messenger like this?*

- Probably my colleagues.

- The instant messenger should be connected with already solved solutions or solution files, a bit like a *Frequently Asked Questions* feature
- A chat like this has to be faster and easier than e-mail and should not distract oneself.
- It would also make sense for new colleagues. If in the contact profiles special knowledge areas of a person are indicated, new colleagues could get along better
- But anyway, I would continue to use my telephone.
- A chat conversation with the support would make sense.
- I do not want to post every problem to an open chat room.

SECOND INTERVIEW *How do you solve issues connected to your work with Solid Edge?*

- In general, we have relatively few problems. Most of the time we do workarounds because a complete problem solution would take too much time. Only if many of our colleagues have the same problem, we contact our support.
- Most of the time I ask colleagues on the same site
- I also use Google and search within discussion boards, but I do not ask new questions because this would take too much time. Time is the most valuable unit in our work.

*What do you think of an integrated instant messenger for Solid Edge? Who would be your contacts in an instant messenger like this?*

- Within one company, a chat for problem solutions maybe reach its limits. And I am not sure if other Solid Edge users have time for that.
- I probably would not take the time to answer problems within a chat
- But I would try out such a chat.
- Screenshot sending is very important, best would be if annotations are allowed. We use the screenshot program *Hardcopy*. Also important is to see if a contact is currently online. Being able to see the phone number would be useful, too.
- A *last used commands* list within a chat similar to the list you get when you click on the back button in Solid Edge would make sense.

THIRD INTERVIEW (TWO CONSTRUCTORS AT ONCE) *How do you solve issues connected to your work with Solid Edge?*

- We ask our colleagues and do not really use the internet. Sometimes we use the built-in Solid Edge help to understand a command better
- I once created a certain workaround for thread production - it took me many hours to figure it out, but now everyone in our department does it this way.
- We only use the support when nobody has an idea - most of the time, these are things which are not directly connected to construction
- Some problems only derive from our old hardware. Trainings for newer versions of Solid Edge are needed too.
- We once took videos to explain a feature better.

*What do you think of an integrated instant messenger for Solid Edge? Would video recording be a useful feature?*

- Taking videos in a chat like this would only be useful when communicating with an external contact. Our support already offers remote-desktop help so this feature would maybe not needed.
- An instant messenger could be helpful in general if it is very simple to use, but who we should contact?
- It could get a problem that people who have more experience with Solid Edge probably will be messaged more often and so loose more time completing their own tasks. In the end, it is again a question about time. It should be possible to *accept* or *not accept* new chat requests

*Other comments?*

- We think a list where all contact persons for a certain section are located would be useful.

#### 9.1.2.3 *Medium-Sized Company in the Injection Moulding Sector, about 60-100 employees*

*How do you work with Solid Edge?*

- I work with Solid Edge version ST5.
- My working hours differs over time. Sometimes I am working with Solid Edge 8h/day for four weeks, then not at all for some time.
- My task is the construction of new parts and project management
- I am the only one in our company who uses Solid Edge

*How do you solve issues connected to your work with Solid Edge?*

- Because I am the only one working with Solid Edge, I have to solve them on my own. Most of the time I'll make it work after trying it for some time.
- In the past I could discuss issues with a colleague, but he has resigned
- In the past I used to call the support more often, nowadays this does not happen that often

*How do you communicate with your support? What information do they need from you?*

- Mostly by phone. Then I send them the respective files per mail and they try it at their own computers.

*What do you think of an integrated instant messenger for Solid Edge?*

- I probably would not use it.
- If I would have to look at my own like in a discussion board if there are problems I can contribute too, I would not have time for that
- If I would get notifications when someone wants specifically my help, I would be more likely to answer. But notifications like this shouldn't appear too frequently.

*Do you have concerns about an integrated instant messenger in terms of security and privacy?*

- No, rather not.

## 9.2 EMBEDDEDCHAT SCREENSHOTS

Some FrontendApplication views were not shown in section 6.3.3 because of their simplicity. To have a complete view on EmbeddedChat, they are shown in this section.

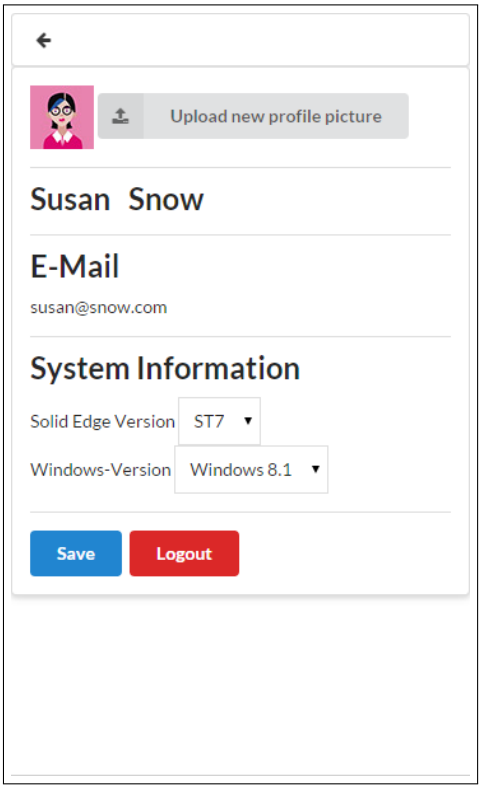



Figure 27: Viewing one's own user profile

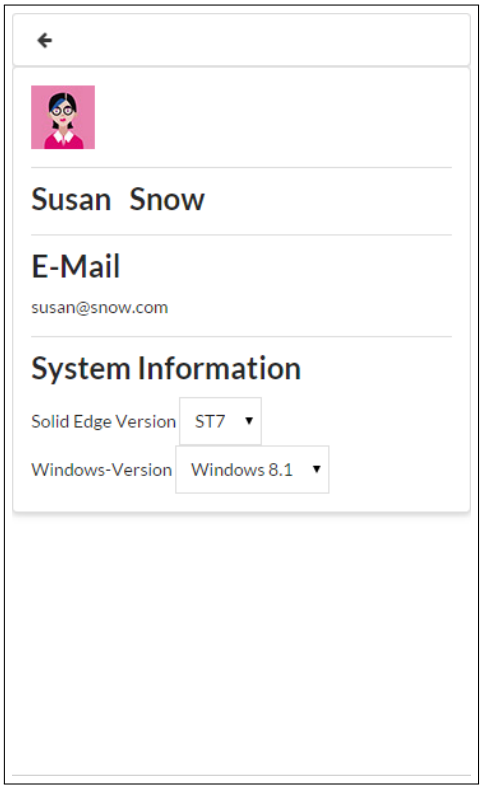



Figure 28: Viewing another user's profile

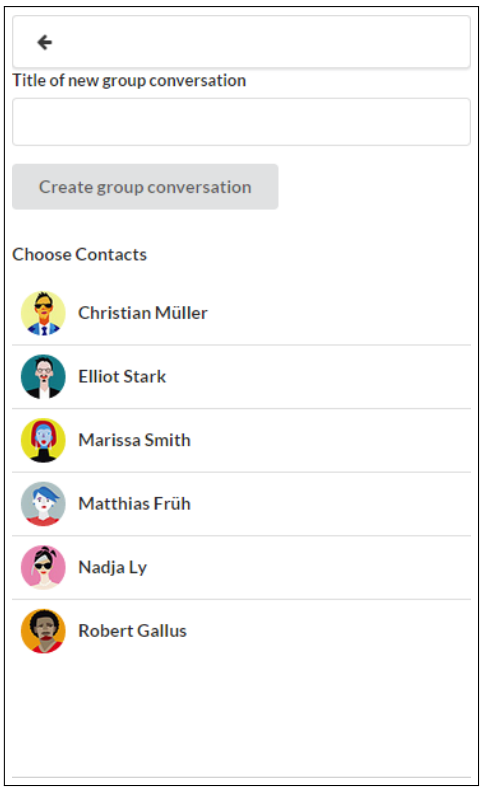



Figure 29: Creating a new group conversation

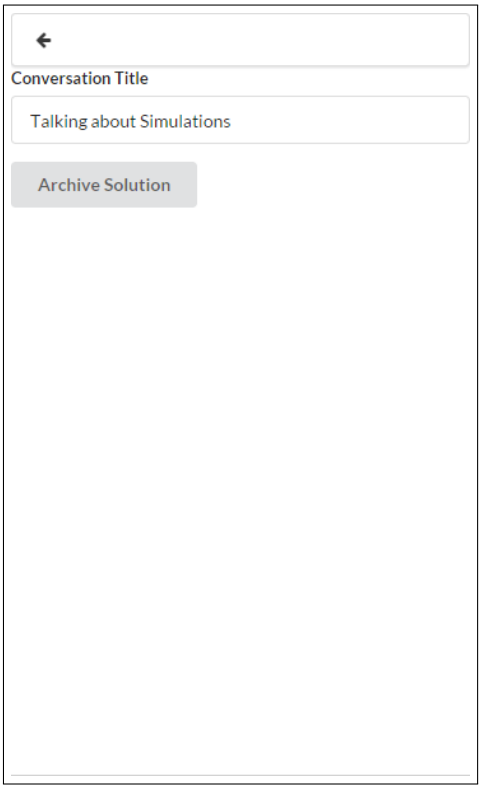



Figure 30: Archiving a conversation

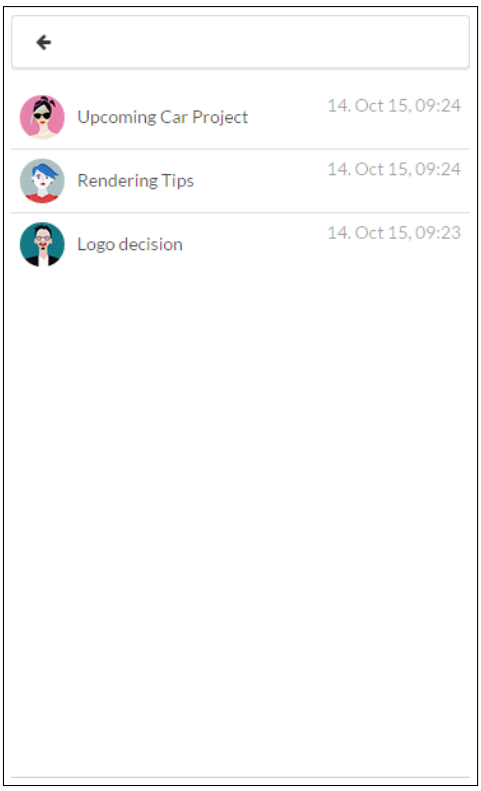



Figure 31: Viewing the list of archived conversations

## 9.3 EMBEDDEDCHAT COMPONENT COMMUNICATION PROTOCOLS

### 9.3.1 *REST-API*

| **HTTP Method** | **URI** | **Description** |
|---|---|---|
| **POST** | /auth/sign_in | Checks the login information of a user and returns a token which the user can use from now on to prove his identity. |
| **GET** | /auth/validate_token | Returns "success" if the user has sent a valid token with his request. |

Table 5: URIs related to Authentication

| | | |
|---|---|---|
| **GET** | /conversations | Returns an array of all conversations. Filtering on "finished" state and "user_id" is possible with URL parameters. |
| **POST** | /conversations/create | Creates a new conversation. |
| **PUT** | /conversations/finish/{conversation_id} | Sets a conversation state to "finished" (=archived). |
| **GET** | /conversations/{conversation_id} | Returns a conversation. |
| **GET** | /conversations{conversation_id}/users | Returns the list of users participating in the conversation. |

Table 6: URIs related to Conversations

| | | |
|---|---|---|
| **GET** | /images/{image_id}/{original?} | Shows an image. The user can specify if he wants to see the original or a downscaled version. |

Table 7: URI related to Images

| | | |
|---|---|---|
| **GET** | /messages/{conversation_id} | Returns an array of all messages (text, images, events) of a conversation. |
| **GET** | /messages/{conversation_id}/newest | Returns the newest message of a conversation. |
| **POST** | /messages/{conversation_id}/send | Adds a new text message to a conversation. |
| **POST** | /messages/{conversation_id}/sendImage | Uploads a new image (max. 5 MB) and returns the new id of the image. |
| **POST** | /messages/{conversation_id}/sendScreenshot | Out of a base64 string, saves a new image and returns the new id. |
| **GET** | /messages/{conversation_id}/{message_id} | Returns the single requested message. |

Table 8: URIs related to Messages

| | | |
|---|---|---|
| **GET** | /users | Returns a list of all users, except the user stated within the optional URL parameter "user_id". |
| **POST** | /users/create | Creates a new user. |
| **GET** | /users/{user_id} | Returns a user with specified user_id. |
| **POST** | /users/{user_id}/change | Updates a user's properties. |
| **GET** | /users/{user_id}/ getSummaryOfNewMessages | Returns the list of new messages since the user was last online. |
| **GET** | /users/{user_id}/profile_picture | Shows the profile picture of a user. |
| **POST** | /users/{user_id}/profile_picture | Uploads a new profile picture (max. 5 MB) for a user. The updated user entity is returned. |

Table 9: URIs related to Users

### 9.3.2 *Socket.IO Defined Event Types*

See table 10 for the list of self-imposed socket.io event types.

| Event Direction | Name | Description |
|---|---|---|
| **Client to Server** | message | This event contains a new text message. The server adds it to the corresponding conversation and broadcasts the message to all online conversation members. |
| | image | This event contains the URL of a previously uploaded (via REST) image. The server adds to the corresponding conversation and broadcasts the image to all online conversation members. |
| | event | This event contains information about an Event Timeline command event which occurred in Solid Edge. The server adds it to the corresponding conversation and broadcasts the command event to all online conversation members. |
| | disconnect | The client signalizes that he will disconnect the connection. The server marks the client as offline and informs the other online clients with a "logout" event. |
| | askwhoonline | The client asks for a list of currently online clients. The server responds with an "logged-in" event. |
| | spread-join-invitation | The clients asks the server to spread an invitation for a new conversation to a list of participants. The server sends a "join-invitation" to these clients. |
| | join-request | A client asks the server to be notified about new messages of a conversation. The server checks if the user is a member of this conversation and adds him to the corresponding "Socket.IO room". |
| **Server to Client** | logout | The server informs all online clients of an disconnecting user. The clients change the user's status to "offline". |
| | login | The server informs all online clients of a newly connected user. The clients change the user's status to "online". |
| | logged-in | The server sends a list of all currently connected clients to all clients. The clients change the status of these users to "online". |
| | join-invitation | The server sends an invitation to a conversation to a client. The client then sends a "join-request" event to the server. |
| | error | The server sends an error with additional information to the client. |
| | broadcast | The server sends a message to the client, which can be a text message, a message containing an image URL to display or a command event. |

Table 10: Event types for Socket.IO connection between FrontendApplication (client) and BackendApplication (server)

## 9.4 ONLINE SURVEY EVALUATION

These are the detailed results of the online survey described in chapter 7. The individual answers are available on request.

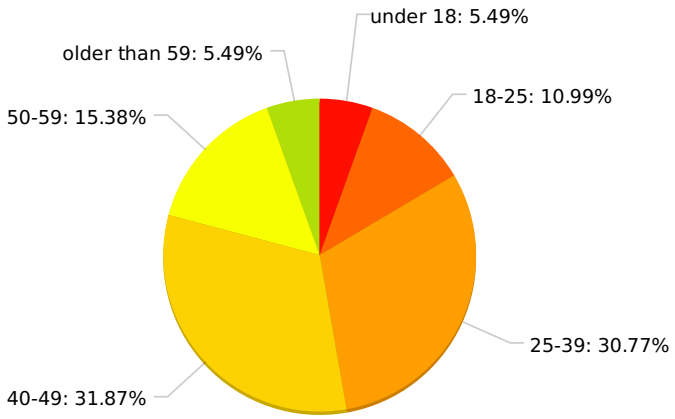



Figure 32: Question 1: Age. Number of participants: 91

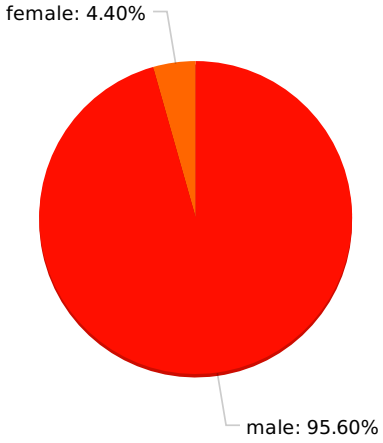



Figure 33: Question 2: Sex. Number of participants: 91

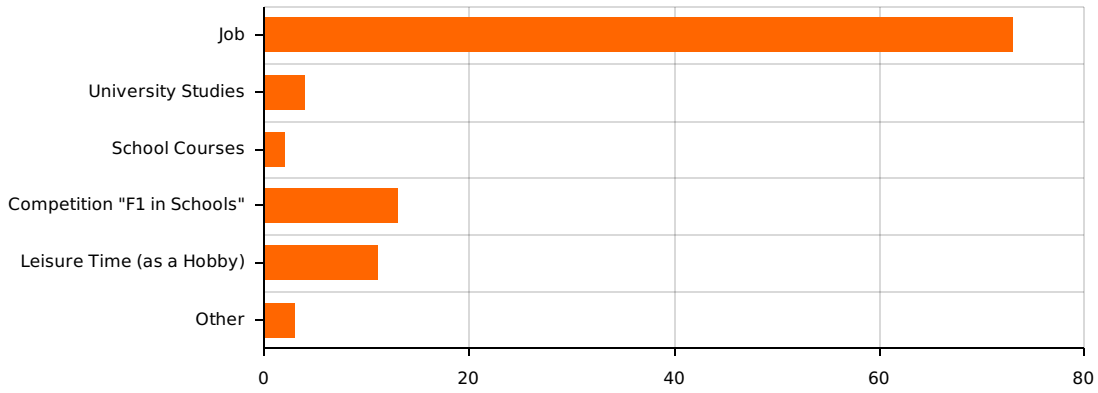



Figure 34: Question 3: In which setting do you use Solid Edge? Number of participants: 91. Other: Reselling, Development

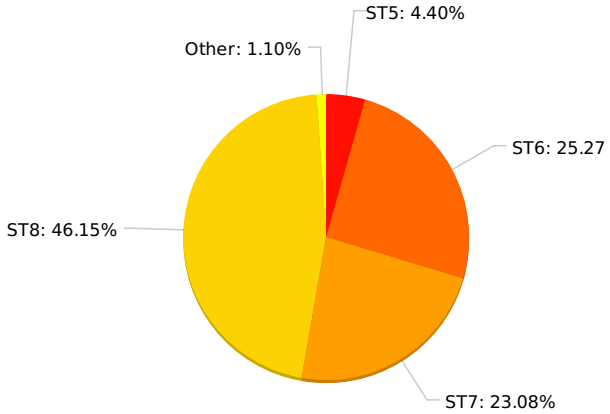



Figure 35: Question 4: Which Solid Edge version are you using? Number of participants: 91. Other: ST6&ST7&ST8

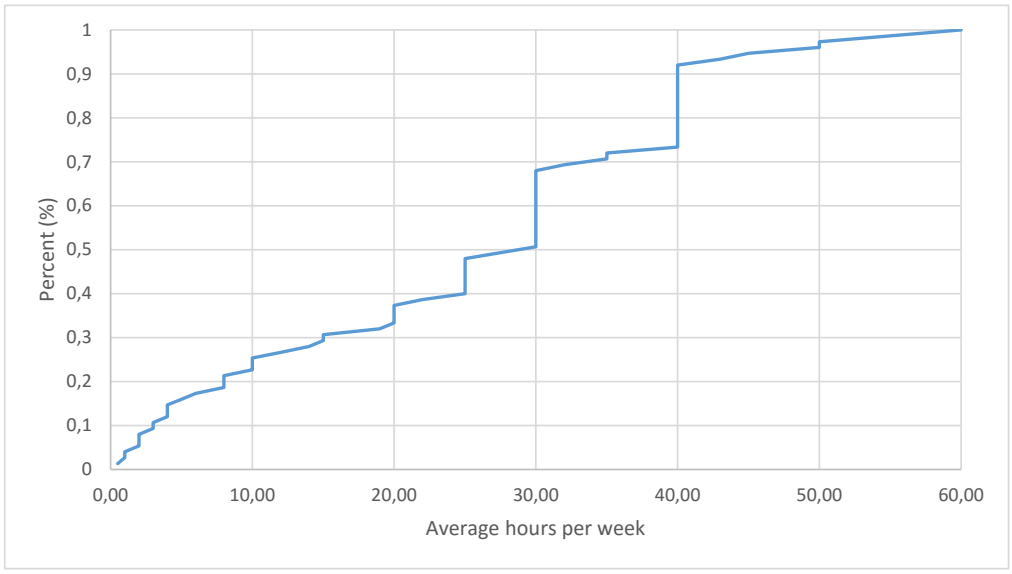



Figure 36: Question 5: How many hours on average per week do you work with Solid Edge? Number of participants: 75

| | 0% (1) | | 10% (2) | | 20% (3) | | 30% (4) | | 40% (5) | | 50% (6) | | 60% (7) | | 70% (8) | | 80% (9) | | 90% (10) | | 100% (11) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Ø | ± |
| Construction of new parts | 11x | 14,47 | 12x | 15,79 | 6x | 7,89 | 10x | 13,16 | 6x | 7,89 | 8x | 10,53 | 8x | 10,53 | 1x | 1,32 | 1x | 1,32 | 5x | 6,58 | 8x | 10,53 | 5,03 | 3,26 |
| Creation of technical dra... | 14x | 18,67 | 20x | 26,67 | 17x | 22,67 | 6x | 8,00 | 2x | 2,67 | 3x | 4,00 | 3x | 4,00 | - | - | 2x | 2,67 | 1x | 1,33 | 7x | 9,33 | 3,77 | 3,05 |
| Modifications on existing ... | 14x | 18,67 | 18x | 24,00 | 13x | 17,33 | 11x | 14,67 | 3x | 4,00 | - | - | 2x | 2,67 | 4x | 5,33 | 5x | 6,67 | - | - | 5x | 6,67 | 3,92 | 3,00 |
| Prototype Development | 34x | 45,95 | 12x | 16,22 | 9x | 12,16 | 3x | 4,05 | 4x | 5,41 | 4x | 5,41 | 2x | 2,70 | - | - | 2x | 2,70 | 2x | 2,70 | 2x | 2,70 | 2,91 | 2,70 |
| Simulation (with Solid Ed... | 49x | 67,12 | 15x | 20,55 | 6x | 8,22 | 2x | 2,74 | 1x | 1,37 | - | - | - | - | - | - | - | - | - | - | - | - | 1,51 | 0,87 |
| Creation of Renderings a... | 48x | 65,75 | 21x | 28,77 | 2x | 2,74 | 1x | 1,37 | - | - | 1x | 1,37 | - | - | - | - | - | - | - | - | - | - | 1,45 | 0,82 |
| Viewing other constructio... | 23x | 31,08 | 28x | 37,84 | 4x | 5,41 | 4x | 5,41 | 6x | 8,11 | 3x | 4,05 | 2x | 2,70 | - | - | - | - | - | - | 4x | 5,41 | 2,88 | 2,52 |
| Giving support to other S... | 29x | 39,73 | 10x | 13,70 | 4x | 5,48 | 4x | 5,48 | 2x | 2,74 | 2x | 2,74 | 3x | 4,11 | - | - | 5x | 6,85 | 2x | 2,74 | 12x | 16,44 | 4,34 | 3,94 |

Figure 37: Question 6: Which tasks do you perform using SolidEdge? How much time do you spend relatively performing each of these tasks? Number of participants: 76. User-added entries were omitted from this diagram.

Some answers in figure 37 are not fully visible, here their full text:

- Creation of technical drawings
- Modifications on existing parts
- Simulation (with Solid Edge Simulation)
- Creation of Renderings and Animations
- Viewing other constructions/parts
- Giving support to other Solid Edge users

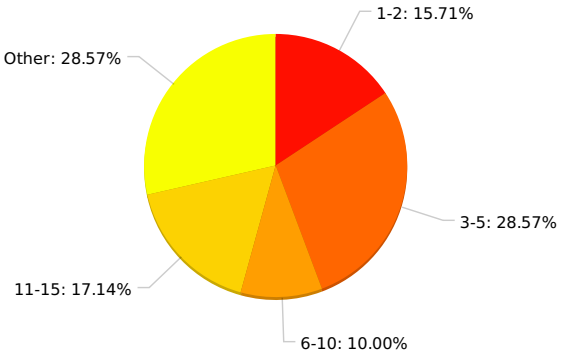



Figure 38: Question 7: With how many persons are you in contact frequently in context of your work with Solid Edge? Number of participants: 76. Other: 80; 20; 100; 100; 20; more; 0; 250; 50; >50; 100; 100; 30; 30; 20; large number; 20; 0; 20; 60

| | several times a day (1) | | daily (2) | | every 2-3 days (3) | | weekly (4) | | every 2-3 weeks (5) | | monthly (6) | | less frequently (7) | | never (8) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Ø | ± |
| Meetings in person | 22x | 31,43 | 17x | 24,29 | 4x | 5,71 | 7x | 10,00 | 4x | 5,71 | 4x | 5,71 | 8x | 11,43 | 4x | 5,71 | 3,26 | 2,37 |
| E-Mail | 21x | 30,00 | 18x | 25,71 | 5x | 7,14 | 8x | 11,43 | 1x | 1,43 | 6x | 8,57 | 4x | 5,71 | 7x | 10,00 | 3,27 | 2,41 |
| Phone | 17x | 24,29 | 17x | 24,29 | 9x | 12,86 | 5x | 7,14 | 3x | 4,29 | 9x | 12,86 | 5x | 7,14 | 5x | 7,14 | 3,46 | 2,33 |
| Facebook or a Social Net... | 3x | 4,29 | 7x | 10,00 | 2x | 2,86 | 6x | 8,57 | - | - | 2x | 2,86 | 4x | 5,71 | 46x | 65,71 | 6,50 | 2,41 |
| Instant Messenger | 6x | 8,57 | 12x | 17,14 | 1x | 1,43 | 3x | 4,29 | 2x | 2,86 | 2x | 2,86 | 1x | 1,43 | 43x | 61,43 | 5,97 | 2,79 |

Figure 39: Question 8: Which means of communication are then used? How often for each means? Number of participants: 70. User-added entries were omitted from this diagram.

One answer in figure 39 is not fully visible, here its full text: Facebook or a Social Network

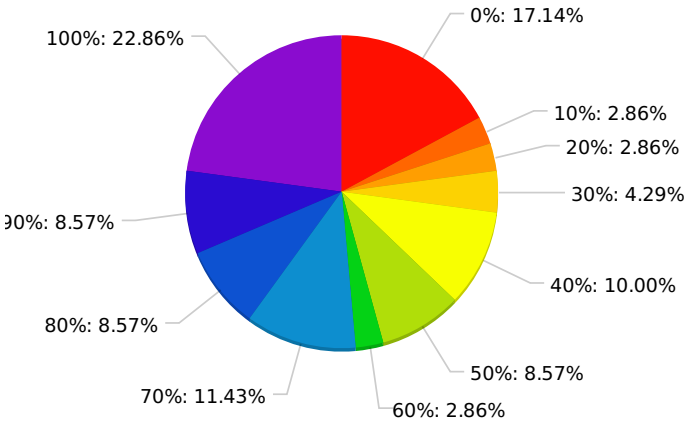



Figure 40: Question 9: Which portion of these contacts uses Solid Edge regularly? Number of participants: 70

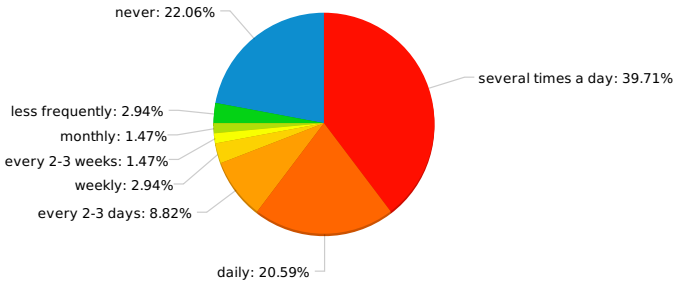



Figure 41: Question 10: How often do you use instant (chat) messengers in your private life? Number of participants: 68

Between question 10 and 11 , the video of EmbeddedChat was shown. It can be found on YouTube. English version: https://www.youtube.com/watch?v=b577ISWLBHI, German version: https://www.youtube.com/watch?v=uAurMs6GLDo

| | Extremely likely (1) | | Quite likely (2) | | Slightly likely (3) | | Neither (4) | | Slightly unlikely (5) | | Quite unlikely (6) | | Extremely unlikely (7) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Ø | ± |
| Using EmbeddedChat in ... | 2x | 3,28 | 13x | 21,31 | 10x | 16,39 | 7x | 11,48 | 9x | 14,75 | 8x | 13,11 | 12x | 19,67 | 4,31 | 1,93 |
| Using EmbeddedChat wou... | 2x | 3,28 | 10x | 16,39 | 8x | 13,11 | 14x | 22,95 | 6x | 9,84 | 7x | 11,48 | 14x | 22,95 | 4,46 | 1,88 |
| Using EmbeddedChat in ... | 1x | 1,64 | 11x | 18,03 | 12x | 19,67 | 10x | 16,39 | 6x | 9,84 | 8x | 13,11 | 13x | 21,31 | 4,39 | 1,87 |
| Using EmbeddedChat wou... | 3x | 4,92 | 7x | 11,48 | 13x | 21,31 | 12x | 19,67 | 5x | 8,20 | 8x | 13,11 | 13x | 21,31 | 4,39 | 1,88 |
| Using EmbeddedChat wou... | 4x | 6,56 | 12x | 19,67 | 10x | 16,39 | 10x | 16,39 | 8x | 13,11 | 6x | 9,84 | 11x | 18,03 | 4,11 | 1,93 |
| I would find EmbeddedCh... | 6x | 9,84 | 13x | 21,31 | 10x | 16,39 | 10x | 16,39 | 4x | 6,56 | 4x | 6,56 | 14x | 22,95 | 4,00 | 2,09 |

Figure 42: Question 11: To what degree do you agree with the following statements after watching the video? Number of participants: 61

Some answers in figure 42 are not fully visible, here their full text:

- Using EmbeddedChat in my job would enable me to accomplish tasks more quickly.
- Using EmbeddedChat would improve my job performance.
- Using EmbeddedChat in my job would increase my productivity.
- Using EmbeddedChat would enhance my effectiveness on the job.
- Using EmbeddedChat would make it easier to do my job.

Table 11: Question 12 (optional): Please elaborate on your answer on "I would find EmbeddedChat useful in my job.". Number of participants: 25. One unsubstantial comment was left out.

| |
|---|
| Sorry but the idea of using "Embedded Chat" is just another distraction from people doing their jobs. Could I see having the occasion to use this... maybe. But the chances that the people I would need to go into this kind of detail with, would NOT be using solid Edge or other CAD programs. So what's the point? |
| embedded Chat ist ein Sicherheitsrisiko und eine Zeitfalle und ich würde das überhaupt nicht nützlich für meinen Beruf finden, wie sich aus der Praxis bei Versuchen auch ergeben hat. |
| Working in the EMEA GTAC Solid Edge Support it would help quite a lot. Today we use Appsharing, avi & screenshots instead |
| möchte nicht gestört werden und keine online-Anbindung an niemand haben. |
| You are not the only company trying to do the same thing.; I already have seen the seem thing from other competitors.; But what I like the most would be the ability to communicate with others without the need to use another program directly.; What I really need is something like a mix of onenote and Skype in solid Edge |
| I am the most experienced Solid Edge user in my group and often answer questions to help others. Having this chat ability would make it easier to help others. |
| z.B. in Kontakt mit einer Servicehotline unseres Resellers |

| |
|---|
| Für Autodidakten sicherlich hilfreich, im tagtäglichen Konstruktionsprozess lenkt dieses System meiner Meinung nach nur von der eigentlichen Arbeit ab. |
| I think EmbeddedChat would be very useful for any Solid Edge user who has to give support to other Solid Edge users. Or for a group of new SE users who want to help each other or collaborate. Probably it will be best used within one company or university, because to connect with other users outside will probably have technical and security difficulties. |
| Most companies already have a corporate chat tool, and in our case it is Cisco Jabber. The attractive part of your tool is the ability to quickly capture the SE window. My suggestion would be to focus on integrating with existing chat tools instead of creating a new one. I'm sure there needs to be a host server to make this work and it is something that would not be allowed here because we already have the capability through Jabber. |
| I can see it being useful for specific actions like described in the video. I can see where it could save a call to GTAC to get a quick answer amoung your pears without leaving the SE interface. We do these same things now with instant messenger and Microsoft Office Communicator. I can see where it being directly in the interface with the ability to quickly create screen captures could be useful. Possibly also to add application sharing where you can show exactly what you are doing live. |
| EmbeddedChat würde beitragen, schnellere und bessere Lösungen und Ergebnisse in Solid Edge zu erzielen - einfach weil man schneller Rat und Hilfe einholen kann, zudem könnten Firmenstandards leichter verbreitet werden. Auf der anderen Seite ist aber sicherlich auch Disziplin erforderlich, um nicht bei jeder Kleinigkeit einen Kollegen zu belästigen - sonst kann es leicht passieren, dass die Kollegen mit dem meisten Know How nicht mehr zum Arbeiten kommen. |
| may help understanding the problem |
| we do not have any external partners who use SE in our industry |
| Currently we use teamviewer in multiple cases so this would make remote connection even easier and probably decrease the email change between customer and me. |
| The major issue as I see it is that this is only capable of static images.; Due to the complexities of what I deal with the interaction with other users needs to be dynamic and interactive.; For example I would need to both have access to the remote users desktop and be able to show my screen in real time. |
| Typically I would use IM for these type of communications - but having a chat integrated directly in Solid Edge would be more efficient |

| |
|---|
| Embedded Chat macht dann Sinn, wenn die traditionellen Kommunikationsmöglichkeiten (pers. Gespräch, Telefonate etc.) scheitern und trotzdem eine Kommunikation benötigt wird.; ; Grundsätzlich aber bringt Embedded Chat (wie alle Kommunikationsmöglichkeiten!) die Gefahr, dass es missbräuchlich genutzt wird.; Dabei ist es egal, ob nun über fachliche oder andere Dinge gechattet wird. In jedem Fall lenkt es die Beteiligten von ihrer eigentlichen Arbeit ab.; Es stellt sich dann die Frage, wie man den Kommunikationsverkehr steuern und überwachen kann und dabei rechtlich noch auf der sicheren Seite ist.; ; Darüber hinaus wird ein 2. Kommunikationskanal geschaffen, was ich insgesamt auch für die innerbetrieblichen Abläufe als kritisch ansehe. Selbst wenn man die missbräuchliche Nutzung mal außer Acht lässt - darauf können ja immer nur Mitarbeiter aus dem CAD-Bereich zugreifen! |
| Die 6 Fragen meinen nach meinem Verständnis mehr oder weniger dasselbe. Daher auch relative gleiche Antworten.; Ich kann mir vorstellen dass speziell bei Problemen die gegenseitige Hilfe einfacher ist, wenn der helfende Kollege nicht im gleichen Raum sitzt. (bzw. ein Support). Man könnte auch Kniffe und Tricks an Kollegen verteilen, oder firmeninterne Festlegungen (Quasi Werks- Konstruktionsnormen). |
| Ich vermute das es mich mehr von der Arbeit abhält, als Zeit einspart.; Bei Fragen zur Bedienung spreche ich meine Kollegen im unmittelbaren Umfeld direkt an. (Sitzen neben mir) |
| Die archivierten Anfragen und Antworten würden einigen Kollegen weiterhelfen, da mehrfach gestellte Fragen somit von Embedded Chat beantwortet werden können und nicht von jemand anderen, der seine Arbeit unterbrechen müsste. |
| Ich arbeite im Solid Edge Support, da wäre es manchmal hilfreich neben diversen Online Meeting Programmen auch wie gezeigt mal einen Screenshot zu übermitteln oder Hilfestellung per Textnachricht zu geben. Ich muß aber auch die Möglichkeit haben den Chat zu deaktivieren. Nur weil ich gerade Solid Edge gestartet habe heißt das nicht das ich den Kopf frei habe. Vielleicht leiste ich auch gerade telefonisch Support oder halte eine Schulung.; Die Eingrenzung auf rein Firmeninterne Teams würde die Akzeptanz in den Firmen sicher erhöhen. Wichtig ist das firmeninterne Kommunikation nicht versehentlich nach außen dringt. |
| In meinem Fall arbeite ich in einem Großraumbüro und gehe lieber persönlich zum jeweiligen Hilfesuchenden Konstrukteur.; Ich bin hier CAD Admin.; Und sollte ich jemandem in weiter ferne helfen, benutze ich lieber Lösungen wie TeamViewer... |
| Wir setzten Sametime von Lotus Notes Konzernweit ein, da ist es häufig nervig von jedem erreichbar zu sein. Hier wäre man durchaus nur unter der Konstruktion erreichbar.... |

| | 1 (very important) (1) | | 2 (2) | | 3 (3) | | 4 (4) | | 5 (5) | | 6 (6) | | 7 (7) | | 8 (absolutely unimportant) (8) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Ø | ± |
| Integration into Solid Edg... | 10x | 16,39 | 6x | 9,84 | 12x | 19,67 | 10x | 16,39 | 1x | 1,64 | 1x | 1,64 | 3x | 4,92 | 18x | 29,51 | 4,49 | 2,68 |
| See Online Status of Cont... | 14x | 22,95 | 13x | 21,31 | 6x | 9,84 | 7x | 11,48 | 2x | 3,28 | 4x | 6,56 | 1x | 1,64 | 14x | 22,95 | 3,92 | 2,69 |
| Send Screenshots | 19x | 31,15 | 14x | 22,95 | 7x | 11,48 | 3x | 4,92 | 2x | 3,28 | 2x | 3,28 | 1x | 1,64 | 13x | 21,31 | 3,49 | 2,72 |
| See recently used comma... | 9x | 14,75 | 5x | 8,20 | 12x | 19,67 | 9x | 14,75 | 5x | 8,20 | 5x | 8,20 | 2x | 3,28 | 14x | 22,95 | 4,46 | 2,47 |
| Archive conversations an... | 10x | 16,39 | 10x | 16,39 | 12x | 19,67 | 5x | 8,20 | 4x | 6,56 | 5x | 8,20 | 1x | 1,64 | 14x | 22,95 | 4,18 | 2,57 |
| Notifications about new ... | 17x | 27,87 | 10x | 16,39 | 9x | 14,75 | 4x | 6,56 | 2x | 3,28 | 4x | 6,56 | 3x | 4,92 | 12x | 19,67 | 3,79 | 2,70 |
| Group Conversations | 12x | 19,67 | 13x | 21,31 | 7x | 11,48 | 8x | 13,11 | 3x | 4,92 | 2x | 3,28 | 2x | 3,28 | 14x | 22,95 | 4,00 | 2,65 |
| Send images | 12x | 19,67 | 13x | 21,31 | 8x | 13,11 | 6x | 9,84 | 3x | 4,92 | 5x | 8,20 | 2x | 3,28 | 12x | 19,67 | 3,95 | 2,58 |

Figure 43: Question 13: How important are the following EmbeddedChat features for you? Number of participants: 61

Some answers in figure 43 are not fully visible, here their full text:

- Integration into Solid Edge as EdgeBar
- See Online Status of Contacts
- See recently used commands of your contacts
- Archive conversations and provide them with a title
- Notifications about new messages

| | Extremely likely (1) | | Quite likely (2) | | Slightly likely (3) | | Neither (4) | | Slightly unlikely (5) | | Quite unlikely (6) | | Extremely unlikely (7) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Σ | % | Ø | ± |
| I would appreciate an int... | 1x | 1,67 | 11x | 18,33 | 17x | 28,33 | 9x | 15,00 | 3x | 5,00 | 5x | 8,33 | 14x | 23,33 | 4,22 | 1,91 |
| It would be easy for me t... | 20x | 33,33 | 25x | 41,67 | 7x | 11,67 | 3x | 5,00 | - | - | 2x | 3,33 | 3x | 5,00 | 2,27 | 1,55 |
| I would have a clear visio... | 13x | 21,67 | 17x | 28,33 | 7x | 11,67 | 9x | 15,00 | 4x | 6,67 | 2x | 3,33 | 8x | 13,33 | 3,20 | 2,01 |
| It would be easy to establ... | 7x | 11,67 | 11x | 18,33 | 7x | 11,67 | 10x | 16,67 | 6x | 10,00 | 6x | 10,00 | 13x | 21,67 | 4,12 | 2,10 |
| It would be easy to persu... | 5x | 8,33 | 9x | 15,00 | 12x | 20,00 | 11x | 18,33 | 4x | 6,67 | 3x | 5,00 | 16x | 26,67 | 4,22 | 2,06 |
| EmbeddedChat would be... | 3x | 5,00 | 8x | 13,33 | 11x | 18,33 | 13x | 21,67 | 5x | 8,33 | 5x | 8,33 | 15x | 25,00 | 4,40 | 1,93 |

Figure 44: Question 14: To what degree do you agree with the following statements? Number of participants: 60

Some answers in figure 44 are not fully visible, here their full text:

- I would appreciate an introduction of EmbeddedChat
- It would be easy for me to learn how to use EmbeddedChat.
- I would have a clear vision for which purpose I can use EmbeddedChat

- It would be easy to establish EmbeddedChat at my workplace.
- It would be easy to persuade my contacts to use EmbeddedChat.
- EmbeddedChat would become an established communication channel.

Table 12: Question 15 (optional): Last question: Is there anything you would like to tell me? Number of participants: 13. Answers containing personal information were left out or anonymized.

| |
|---|
| The “slightly likely” answer above is due to firewalls, proxy, company security etc |
| Like I said before, I think what we really need is something like Onenote & skype integrated in SE |
| As I said previously, companies with an established corporate chat may be quite resistant to another chat tool, so being able to incorporate existing chat clients might be a better idea in those cases. I believe there is a chat standard that allows this type of integration. |
| Persönlich finde ich Embedded Chat als eine tolle Idee, jedoch ist ein Kommunikationskanal bei einigen Mitarbeitern eher kontraproduktiv.; Es gibt leider immer wieder Kollegen bzw. Mitarbeiter die solche Medien nicht nur für Problemstellungen benutzen sondern eher zur allgemeinen Kommunikation. Trotzdem würde ich dieses Tool sehr begrüßen, da man sicherlich mit einer sinnvollen Archivierung oder einer evtl. einsehbaren Archivierung die “falsch Nutzung” von Embedded Chat unterbinden bzw. eingrenzen könnte. |
| grundsätzlich eine interessante Lösung. Für uns als mittelständisches Unternehmen ist eine alleinige Solid Edge Lösung leider nicht optimal.; Wir haben viele Anwender im Haus, die nicht ständig Solid Edge offen haben (Projektierung, Vertrieb). Eine Kommunikation wäre dann nicht möglich!; Für Mitarbeiter ohne ständige Solid Edge Nutzung muss ein Lösung gefunden werden. ; Ich könnte mir noch weitere Funktionen vorstellen ... (bin auch selbst Entwickler) |
| Die Funktionalität ist sicher interessant.; Ich könnte mir vorstellen das diese z.B. in die Solid Edge Datenverwaltung Insight, die auf SharePoint basiert mit integriert werden kann.; Dadurch könnten nicht nur Benutzer miteinander kommunizieren auch das System könnte z.B. den Benutzer über wichtige Vorgänge direkt informieren |
| Ich fand die einbindung von Youtube und Facebook als total überflüssig und so ist es auch mit einer Chaterweiterung! |
| mit so einem Chat würde doch keiner zum arbeiten kommen !!!!! |

# BIBLIOGRAPHY


Christopher Allen. *Life with alacrity: Tracing the evolution of social software*. 2004. URL http://www.lifewithalacrity.com/2004/10/tracing_the_evo.html. Accessed on 1.09.2015. (Cited on pages 1 and 5.)

Amazon.de. Hilfe: Mayday: Hilfe für das Fire-Tablet erhalten. URL http://www.amazon.de/gp/help/customer/display.html?nodeId=201540070. Accessed on 10.10.2015. (Cited on page 16.)

angular translate. i18n for your Angular apps, made easy. URL https://angular-translate.github.io/. Accessed on 10.10.2015. (Cited on page 25.)

AngularJS. URL https://angularjs.org/. Accessed on 10.10.2015. (Cited on pages 22 and 25.)

Autodesk. Cloud-Service | Design in der Cloud | Autodesk 360. URL http://www.autodesk.de/360-cloud. Accessed on 10.10.2015. (Cited on page 12.)

Autodesk. Use Communicator for Revit, September 2015. URL http://knowledge.autodesk.com/support/revit-products/getting-started/caas/CloudHelp/cloudhelp/2015/ENU/Revit-CAR/files/GUID-36BC2739-D565-46DE-8750-B0396CFD0376-htm.html. Accessed on 10.10.2015. (Cited on page 13.)

Autodesk Building Solutions. Autodesk A360 Collaboration for Revit: Using Communicator for Revit, February 2015. URL https://www.youtube.com/watch?v=wg4y4eYEFIc. starting from minute 25:38; accessed on 10.10.2015. (Cited on page 13.)

Balsamiq. URL https://balsamiq.com/. Accessed on 10.10.2015. (Cited on page 25.)

Michael Bächle. Social Software. *Informatik-Spektrum*, 29(2):121–124, March 2006. ISSN 0170-6012, 1432-122X. doi: 10.1007/s00287-006-0063-2. URL http://link.springer.com/article/10.1007/s00287-006-0063-2. (Cited on page 1.)

Erin Bradner, Wendy a. Kellogg, and Thomas Erickson. The Adoption and Use of 'BABBLE': A Field Study of Chat in the Workplace. *Proceedings of the Sixth European Conference on Computer Supported Cooperative Work 12–16 September 1999, Copenhagen, Denmark*, (September):139–158, 1999. doi: 10.1007/978-94-011-4441-4. URL http://www.springerlink.com/index/10.1007/978-94-011-4441-4. Accessed on 1.09.2015. (Cited on page 14.)

Vannevar Bush. As We May Think. *The atlantic monthly*, 176(1):101–108, 1945. (Cited on page 1.)

CAD.de. Solid Edge. URL http://forum.cad.de/cgi-bin/ubb/forumdisplay.cgi?action=topics&forum=Solid+Edge&number=3. Accessed on 10.10.2015. (Cited on page 18.)

CefSharp. .NET (WPF and Windows Forms) bindings for the Chromium Embedded Framework. URL https://github.com/cefsharp/CefSharp. Accessed on 10.10.2015. (Cited on pages 35 and 46.)

Per Christensson. What's the difference between e-mail, online chat, and instant messaging?, 17.9.2004. URL http://pc.net/helpcenter/answers/email_chat_and_instant_messaging. Accessed on 9.10.2015. (Cited on page 5.)

ClipDis. URL http://www.clipdis.com/. Accessed on 10.10.2015. (Cited on page 8.)

Josh Constine. Facebook Launches Messenger Platform For Content Tools And Chat With Businesses, March 2015. URL http://social.techcrunch.com/2015/03/25/facebook-launches-messenger-platform-with-content-tools-and-chat-with-businesses/. Accessed on 10.10.2015. (Cited on page 8.)

Crew. Slack vs. HipChat. URL http://slackvshipchat.com/?og=1. Accessed on 10.10.2015. (Cited on page 11.)

Fred D. Davis. Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3):319–340, 1989. ISSN 02767783. doi: 10.2307/249008. (Cited on pages 3, 40, and 41.)

F1 in Schools. URL http://www.f1inschools.com/. Accessed on 9.10.2015. (Cited on page 5.)

R.T. Fielding and R.N. Taylor. Principled design of the modern Web architecture. *Proceedings of the 2000 International Conference on Software Engineering. ICSE 2000 the New Millennium*, 2(2):115–150, 2000. ISSN 0270-5257. doi: 10.1109/ICSE.2000.870431. (Cited on page 34.)

GeckoFX. URL https://bitbucket.org/geckofx/. Accessed on 10.10.2015. (Cited on pages 35 and 46.)

Caren Stella Geiger. WhatsApp Calls aktivieren: Neues Telefon-Feature für Android ist da, March 2015. URL http://www.chip.de/news/WhatsApp-Calls-aktivieren-Neues-Telefon-Feature-fuer-Android-ist-da_67919239.html. Accessed on 10.10.2015. (Cited on page 7.)

GitHub. SolidEdgeCommunity/SolidEdge.Community.AddIn. URL https://github.com/SolidEdgeCommunity/SolidEdge.Community.AddIn. Accessed on 10.10.2015. (Cited on page 35.)

Robert Gleixner. *Implementation of Collaboration Features in CAD Software*. Master, Technische Universität München, 2015. (Cited on pages 1, 19, and 22.)

Google Apps for Work. Einfache und flexible Tarife. URL www.google.com/intl/de/work/apps/business/pricing.html. Accessed on 10.10.2015. (Cited on page 10.)

Jessica Guynn. Facebook's WhatsApp hits 900 million users, aims for 1 billion, September 2015. URL http://www.usatoday.com/story/tech/2015/09/04/whatsapp-facebook-900-million-mark-zuckerberg-jan-koum-messenger/71704760/. Accessed on 10.10.2015. (Cited on page 6.)

hapi.js. URL http://hapijs.com/. Accessed on 10.10.2015. (Cited on page 32.)

Dick Hardt and Michael Jones. The OAuth 2.0 Authorization Framework: Bearer Token Usage. URL https://tools.ietf.org/html/rfc6750. (Cited on page 34.)

James D Herbsleb, David L Atkins, David G Boyer, Mark Handel, and Thomas A Finholt. Introducing instant messaging and chat in the workplace. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 171–178. ACM, 2002. (Cited on pages 14, 20, and 45.)

Heroku. URL https://www.heroku.com/. Accessed on 10.10.2015. (Cited on page 37.)

Lachmi Khemlani. Autodesk University 2014: AECbytes Newsletter, December 2014. URL http://www.aecbytes.com/newsletter/2014/issue_72.html. Accessed on 10.10.2015. (Cited on pages ix and 13.)

Nigel King, C. Cassell, and G. Symon. Qualitative methods in organizational research: A practical guide. *The Qualitative Research Interview*, 1994. URL https://perswww.kuleuven.be/~u0035056/PhD/samenvattingen/theories/interviewing/Nigel%20King.doc. (Cited on page 17.)

Bill Knittle. BIM Collaboration Beyond Revit Revisited: Welcome Autodesk A360 Collaboration for Revit and A360 Team, March 2015. URL http://synergiscadblog.com/2015/03/04/bim-collaboration-beyond-revit-revisited-welcome-autodesk-a360-collaboration-for-revit-and-a360-team/. Accessed on 10.10.2015. (Cited on page 13.)

John Koetsier. Flickr founder Stewart Butterfield's new Slack signed up 8,000 companies in 24 hours, August 2013. URL http://venturebeat.com/2013/08/15/flickr-founder-stewart-butterfields-new-slack-signed-up-8000-companies-in-24-hours/. Accessed on 10.10.2015. (Cited on page 11.)

Robert McMillan. You May Not Use WhatsApp, But the Rest of the World Sure Does | WIRED, 20.02.2014. URL http://www.wired.com/2014/02/whatsapp-rules-rest-world. Accessed on 10.10.2015. (Cited on page 6.)

Microsoft. WebBrowser Class (System.Windows.Forms). URL https://msdn.microsoft.com/en-us/library/system.windows.forms.webbrowser(v=vs.110).aspx. Accessed on 10.10.2015. (Cited on page 35.)

Markus Müller. Bachelorarbeit-Umfrage Chat-Erweiterung für Solid Edge, September 2015. URL http://forum.cad.de/foren/ubb/Forum3/HTML/011693.shtml. Accessed on 10.10.2015. (Cited on page 40.)

MongoDB. URL https://www.mongodb.org/. Accessed on 10.10.2015. (Cited on pages 22 and 34.)

Mongoose. URL http://mongoosejs.com/. Accessed on 10.10.2015. (Cited on page 34.)

Michael J. Muller, Mary Elizabeth Raven, Sandra Kogan, David R. Millen, and Kenneth Carey. Introducing Chat into Business Organizations: Toward an Instant Messaging

Maturity Model. In *Proceedings of the 2003 International ACM SIGGROUP Conference on Supporting Group Work*, GROUP '03, pages 50–57, New York, NY, USA, 2003. ACM. ISBN 1-58113-693-5. doi: 10.1145/958160.958168. URL http://doi.acm.org/10.1145/958160.958168. (Cited on pages 14 and 15.)

Bonnie A. Nardi, Steve Whittaker, and Erin Bradner. Interaction and Outeraction: Instant Messaging in Action. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, CSCW '00, pages 79–88, New York, NY, USA, 2000. ACM. ISBN 1-58113-222-0. doi: 10.1145/358916.358975. URL http://doi.acm.org/10.1145/358916.358975. (Cited on page 4.)

Jason Newell. Solid Edge ST7 AddIn Architecture Overview, 11.10.2014. URL http://www.codeproject.com/Articles/839585/Solid-Edge-ST-AddIn-Architecture-Overview. (Cited on page 35.)

Jason Newell. Spy for Solid Edge, February 2014. URL https://solidedgespy.codeplex.com/Wikipage?ProjectName=solidedgespy. Accessed on 10.10.2015. (Cited on page 37.)

Casey Newton. More than a million people are using Slack every day now, June 2015. URL http://www.theverge.com/2015/6/24/8836087/slack-1-million-daily-users. Accessed on 10.10.2015. (Cited on page 11.)

Node.js. URL https://nodejs.org/en/. Accessed on 10.10.2015. (Cited on pages 22 and 32.)

Parmy Olson. Exclusive: The Rags-To-Riches Tale Of How Jan Koum Built WhatsApp Into Facebook's New $19 Billion Baby, February 2014. URL http://www.forbes.com/sites/parmyolson/2014/02/19/exclusive-inside-story-how-jan-koum-built-whatsapp-into-facebooks-new-19-billion-baby/. Accessed on 10.10.2015. (Cited on page 6.)

Oxford Dictionary. chat - definition of chat in English. URL http://www.oxforddictionaries.com/definition/english/chat. Accessed on 9.10.2015. (Cited on page 4.)

Carly Page. Facebook's Whatsapp buy is a privacy nightmare for users, but it makes sense for the social network, February 2014. URL http://www.theinquirer.net/inquirer/opinion/2329985/facebooks-whatsapp-buyout-is-a-privacy-nightmare-for-users-but-it-makes-sense-for-the-social-network. Accessed on 10.10.2015. (Cited on page 7.)

Sarah Phillips. A brief history of Facebook. *The Guardian*, July 2007. ISSN 0261-3077. URL http://www.theguardian.com/technology/2007/jul/25/media.newmedia. (Cited on page 8.)

PR Newswire. Facebook Reports First Quarter 2015 Results, April 2015. URL http://www.prnewswire.com/news-releases/facebook-reports-first-quarter-2015-results-300070539.html. Accessed on 10.10.2015. (Cited on page 8.)

Margaret Rouse. What is CAD (computer-aided design)? - Definition from WhatIs.com, March 2011. URL http://whatis.techtarget.com/definition/CAD-computer-aided-design. Accessed on 9.10.2015. (Cited on page 5.)

Semantic UI. URL http://semantic-ui.com/. Accessed on 10.10.2015. (Cited on pages 22 and 25.)

Siemens. Solid Edge ST7 SDK. URL http://support.industrysoftware.automation.siemens.com/training/se/107/api/webframe.html. Accessed on 10.10.2015. (Cited on page 35.)

Slack. Be less busy, a. URL https://slack.com/is. (Cited on page 11.)

Slack. Getting Started, b. URL https://slack.com/getting-started/admins. Accessed on 10.10.2015. (Cited on page 11.)

Slack. Integrations, c. URL https://slack.com/services/new. Accessed on 10.10.2015. (Cited on page 12.)

Slack. Pricing Guide, d. URL https://slack.com/pricing. Accessed on 10.10.2015. (Cited on page 11.)

Socket.IO. URL http://socket.io/. Accessed on 10.10.2015. (Cited on page 33.)

Solid Edge Forum. URL http://community.plm.automation.siemens.com/t5/Solid-Edge-Forum/bd-p/solid-edge-forum. Accessed on 10.10.2015. (Cited on page 18.)

Stack Exchange. How do I verify that WhatsApp is using end-to-end encryption?, April 2015. URL http://security.stackexchange.com/questions/79070/how-do-i-verify-that-whatsapp-is-using-end-to-end-encryption. Accessed on 10.10.2015. (Cited on page 7.)

Statista. Facebook Messenger: monthly active users 2015, 2015a. URL http://www.statista.com/statistics/417295/facebook-messenger-monthly-active-users/. Accessed on 10.10.2015. (Cited on page 8.)

Statista. Internet chats and instant messengers: usage frequency in Germany 2014, 2015b. URL http://www.statista.com/statistics/428811/internet-chats-and-instant-messengers-usage-frequency-germany/. Accessed on 10.10.2015. (Cited on pages ix and 4.)

Statista. Most used instant messaging apps 2015, 2015c. URL http://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps/. Accessed on 9.10.2015. (Cited on page 6.)

Statista. WhatsApp: mobile usage penetration in selected countries 2014, 2015d. URL http://www.statista.com/statistics/291540/mobile-internet-user-whatsapp/. Accessed on 10.10.2015. (Cited on page 6.)

Mike Wasson. ASP.NET - Single-Page Applications: Build Modern, Responsive Web Apps with ASP.NET, November 2013. URL https://msdn.microsoft.com/en-us/magazine/dn463786.aspx. Accessed on 10.10.2015. (Cited on page 25.)

WhatsApp. Is my device supported? URL http://www.whatsapp.com/faq/en/general/20951556. Accessed on 10.10.2015. (Cited on page 7.)

WhatsApp. WhatsApp Web, January 2015. URL http://blog.whatsapp.com/614/WhatsApp-Web. Accessed on 10.10.2015. (Cited on page 7.)

Jack White. Autodesk announce A360 Collaboration for Revit | BIMcrunch, December 2014. URL http://bimcrunch.com/2014/12/autodesk-announce-a360-collaboration-revit/. Accessed on 10.10.2015. (Cited on page 12.)

Wikipedia. Base64, October 2015a. URL https://en.wikipedia.org/w/index.php?title=Base64&oldid=685003739. Accessed on: 13.10.2015. (Cited on page 36.)

Wikipedia. Google Docs, Sheets, and Slides, October 2015b. URL https://en.wikipedia.org/w/index.php?title=Google_Docs,_Sheets,_and_Slides&oldid=684641614. Accessed on 10.10.2015. (Cited on page 10.)

Wikipedia. Solid Edge, July 2015c. URL https://de.wikipedia.org/w/index.php?title=Solid_Edge&oldid=144027826. Accessed on 10.10.2015. (Cited on page 5.)

Chester Wisniewski. WhatsApp's privacy investigated by joint Canadian-Dutch probe, January 2013. URL https://nakedsecurity.sophos.com/2013/01/29/whatsapps-privacy-investigated-by-joint-canadian-dutch-probe/. (Cited on page 7.)

Brian York. Slack Is The Most Important Communication Tool In The World, July 2015. URL https://medium.com/@brianyork/slack-is-the-most-important-communication-tool-in-the-world-7518927539a6. Accessed on 10.10.2015. (Cited on page 11.)

Monika Zwettler. Siemens PLM Software launcht Solid Edge ST8, 10.06.2015. URL http://www.konstruktionspraxis.vogel.de/siemens-plm-software-launcht-solid-edge-st8-a-492007/. Accessed on 9.10.2015. (Cited on page 5.)