Andreas Wöls, BSc

Reflection on Task & Time Management: A Conversational Agent Supporting Behavioural Change

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master’s degree programme: Software Engineering and Management

submitted to

Graz University of Technology

Supervisor

Ass.Prof. Dipl.-Ing. Dr.techn. Viktoria Pammer-Schindler

Know-Center Graz

Research Center for Data-Driven Business & Big Data Analytics

Graz, December 2019

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Graz, December 2019 .....

Acknowledgements

At this point, I want to thank all those who have supported me during my degree course and the writing of this master thesis.

To begin with, I want to thank my supervisor, Ass. Prof. Dipl.-Ing. Dr.techn. Viktoria Pammer-Schindler for constructive feedback.

Furthermore, I want to thank my fellow students and my dear friends. Besides countless hours of project work, my friends always encouraged me as a student.

I want to thank my girlfriend, who supported me greatly, in a challenging time with very limited time resources.

Ultimately, I want to thank my family, especially my parents Paula and Johann, who supported me during the study course in the best way imageable. Without the help and the care of you, I would not have experienced so many valuable lessons of life in the last few years.

Abstract

In this master thesis, the design and implementation of a conversational agent and the execution of a field study regarding the domain of Task & Time management and fragmentation are presented.

The expanding global competition and the steady demand for services and products have increased the interest in capabilities of planning, prioritizing and achieving goals for corporations. Employees in companies, especially in software businesses, are experiencing increasing requirements in project quantity, size, and complexity. The purpose of this master thesis is to examine the Task & Time management of employees in a real-life business context. By implementing a conversational agent, reflection about task-related activities was encouraged within a target group. A field study was conducted with ten participants, that investigated the participant's awareness over Time & Task management as well as Time fragmentation. Accompanied by an application tracking tool, participants recorded their time resources and their fragmentation. The results of the field study within the target development company, the time fragmentation of the software engineers and recommendations for conversational agents are discussed in this master thesis.

Table of Content

1. Introduction

Software corporations are confronted with fast technology advances, growing requirements in project quantity, size, and complexity. In this context, Task & Time management has become more and more valuable for software engineers, which are responsible for the creation, adaptation, and maintenance of executable software.

Corporations search for healthy, satisfied and high-performing employees to compete successfully in the market place, where time has become a precious resource.

The necessity for managing short-term and long-term goals is a crucial challenge in order to be successful in project-based working activities in a prosperous manner. The employees in software and knowledge technology companies are anticipated to have a self-organized, productive estimation of their work assessment. Engineers can be supported in their behaviour of task planning and achievement by emerging technologies. Conversational agents have attracted increasing attention over the past few years by supporting behavioural change. The interaction between humans and conversational agents, coupled with a reflection-encouraging design, can positively influence the Task & Time management behaviours of software engineers.

The objective of this master thesis is to examine the Task & Time management of employees in a real-life business context. By implementing a conversational agent, reflection about personal activities was encouraged within a target group. A field study was conducted with 10 participants, that investigated the participant's awareness over Time & Task management and fragmentation. Supplemented by a constantly tracking activity tool, participants recorded their time resources and application interruptions.

1.1 Thesis Overview

The goal of this master thesis is to examine the current situation in a consulting company by implementing a technology-enhanced learning application about Task & Time management and measure the results within a 3-week field study. Reflector, a conversational agent, was provided to employees in order to reflect on personal task planning, prioritization and task achievement.

1.2 Scientific and Practical Relevance

1. Scientific relevance: Reflection-based learning and the influence of Task & Time management on performance is belonging to a developing research area. The study contributes to a better understanding of this area and points out the current topics of interest

2. Practical relevance: The knowledge of building a chatbot with SAP Conversational AI framework, is applied in further related projects within the target company. Task & Time management represents a significant capability for software engineers. By interacting with a conversational agent, the employees have been encouraged to monitor their time within a workday healthy for the company and more importantly for themselves.

1.3 Thesis Structure

The structure of this master thesis is as follows. Chapter 2 introduces the domain of this master thesis in more detail and lays out the related background literature. Substantial and relevant aspects in the context of conversational agents are described additionally. In chapter 3, the research approach is explained. In chapter 4, the procedure of creating a preliminary design concept is presented, including a validation study for the concept. In chapter 5, the steps towards the final implementation of the design concept and the features of Reflector, the implemented conversational agent, are discussed. In chapter 6, the field study with employees in a software engineering and consulting company is displayed. The contents of this chapter include the research questions, tools that have been used, the study context, and the participants as well as study procedure and data analysis. In

chapter 7, the results of the field study are portrayed. Chapter 8 covers the interpretation and conclusion of this study.

2. Background and Related Work

2.1 Task & Time Management

Classic resources can be consumed, managed, and eventually reproduced. However, time is a finite resource and an inaccessible factor, that humans cannot access undefinedly, but spend in a wise and meaningful manner. In the last decades, the literature towards Task & Time management has been attracted many researchers and public interest, mainly because of the expanding global competition and the constant demand for services and products, according to Orlikowsky and Yates (2002).

James T. McCay has made one of the oldest researches on time in his book "The management of Time "(McCay, 1968). Several books appeared afterwards, such as "How to Get Control of Your Time and Your Life "(Lakein, 1973) or "The effective Executive "(Drucker, 1967), which refers to work planning. Lakein wrote about Task & Time management behaviours of graduate and undergraduate student in his book "How to Get Control of Your Time and Your Life".

At the beginning of this emerging research area, the target groups have often been students or highly paid professions. Meanwhile, employees of all kinds of professions, as well as part-time workers, are the focus of selected research methods like experiments, diaries, or self-report questionnaires.

Activities are the elementary parts of task achievement and the core of this work. In general, they can be described as patterns to achieve an overarching object. (Nardi, 1996) The subject and the object of an activity build up the definition. For creating or manipulating an objective, a single or a series of actions is required. Without subjects acting or objects to achieve, the activity is incomplete. The activity provides the subject and object with a definite shape and connects them. Operations appear in a specific situation or circumstance with mediating aspects such as tools. Generally, tools may be physical like, but also conceptual, like a mathematical formula or instructions, principles, or algorithms. Working with tools and mediating activities changes the fundamental properties of an activity.

In the literature, various definitions of Task & Time management occur. In "Students planning in the process of self-regulated learning", Task & Time management is considered as a way of monitoring and controlling time. (Eilam & Aharon, 2003) Within the study "Effects of Time-Management Practices on College Grades", the authors write about Task & Time management as "practices intended to maximize intellectual productivity" (Britton & Tesser, 1991). In "Time Management: Test of a Process Model", it is defined as techniques for managing time. (Macan, 1994) Claessens et al. (2007) define Task & Time management as "behaviour that aims at achieving effective use of time while performing certain goal-directed activities".

Amabile et al. (2011) focused on the progress of working achievement and feelings of accomplishment as well as meaning in work. The authors examined employees at work and daily reports, that might be increasing meaning in work.

The literature of Task & Time management also regards to business performance. Many studies have been conducted to examine the influence of methods or techniques for improving work performance or productivity in general. In" Time Management: Test of a Process Model", the authors tried to find a link of Task & Time management and job performance. The results have been that Task & Time management behaviour is not mediating job performance. However, a direct influence of Task & Time management on job performance was modest in other studies (Skinner, 2005) (Eerde, 2003).

Macan (1994) presented a model for Task & Time management that includes a mediating variable for setting goals and priorities as well as mechanics of t Task & Time management. These two behaviours, ultimately, result in a perceived control of time, which is a mediating variable to job-induced somatic tension and job satisfaction. Macan’s results show us that using Task & Time management techniques helps us to assess a reasonable amount of workload within workday times. The experienced sentiment has a positive impact on job satisfaction, at the same time reducing job-induced and somatic stress.

2.2 Task Fragmentation

Time can be measured, kept, and indicated very well. Time tracking tools for recording working hours, sports, or other daily habits are more popular than ever before. Examples are weight or diet tracker, sleep tracking applications, which are recording the quality, duration, and phases of sleep. IoT-devices with appropriate hardware can monitor our health parameters. In the working area, many companies want their employees to track their time during work precisely. Moreover, students in universities are advised to have an exact time plan in order to complete their semester successfully. The many possibilities of advancing technologies such as IoT-devices or smart glasses are accelerating this trend of self-tracking.

The state-of-the-art literature on task fragmentation has grown over the past years. Authors such as Mark et. al. (2008) contributed to the research of context switching, stopping and resuming to working activities.

In the world of information workers, the daily life consists of many kinds of activities and diverse contexts of operation. Information workers often work on multiple projects and have multiple responsibilities. Mark et al. (2008) investigated time interruptions regarding the context of interruptions. They determined that people complete interrupted activities at the same speed and the same quality. However, the drawback is a higher level of stress, higher frustration, and time pressure.

“A certain amount of interruptions may be tolerable because people can compensate with a higher working speed.” Their result also suggests that interruptions force people to switch work routines and mental states. Furthermore, after a short period of interrupted performance, people react with significantly higher stress, frustration, and pressure.

People in the workplace are interrupted multiple times. The interruption compels the employee to stop his thoughts, and his focus on a specific activity is lost. During this period, the work does not continue, and that implies a loss of productivity.

According to “No Task Left Behind? Examining the Nature of Fragmented Work” fragmentation has two features: the duration of an activity and the frequency of interruption. (Mark et al., 2005)

In the workplace context, employees are faced with interruptions and changing the mindset. According to studies, readopting is time-consuming and requires up to 15 minutes (Horvitz & Iqbal, 2007). In this study, computer users are observed during multitasking and the focus on task suspension and resumption. They tracked user’s computer interaction, measuring the disruption and resumption of software applications. Additionally, they recorded the instant messaging and email-client notifications as well as the active operating system application window. The result of monitoring of 27 users over two weeks was that:” participants spent on average nearly 10 minutes on switches caused by alerts and spent on average another 10 to 15 minutes (depending on the type of interruption) before returning to focused activity on the disrupted task.”

Computer manufacturers have noticed the necessity of using multiple applications on the same device. Personal computers have been improved rapidly to fulfil these requirements with more powerful and faster soft- & hardware. Software engineers have numerous displays enabling the view on more than one application to compare, switch, and work on source code quickly.

The source of interruptions can be distinguished into internal and external sources. Software developers, but also knowledge workers are affected by increasing collaborative practices and increasing productivity by sharing information and communication platforms for this purpose. (Czerwinski et al., 2004)

Mark et al. (2005) examined the influence of context switching. They report that users switch context many times, and 57% of their activities are naturally interrupted.

The research of Pammer et al. (2013) found during a study over 6 weeks with 20 participants that “The value of self-tracking lies in gaining self-knowledge”. Participants, who would “gain insight on their Task & Time management, decided on and implemented reasonable changes.”

2.3 Reflective behaviour

Reflection can be defined as the examination of experiences in the past to fully comprehend an objective. The examination is directed on thoughts, feelings, or mental states. Reflection processes are useful for self-learning on a specific topic, and it is critical for knowledge worker's ability to understand and interpret intricate patterns. Reflection improves confidence in order to achieve certain goal-directed activities (Di Stefano et al., 2014).

Gibbs' cycle of reflection is one of the most popular models that help in understanding intricate patterns. It was designed in 1988 by Graham Gibbs to improve learning from activities in the past. The design is cyclic, where a learning experience is revisited step by step. (The University of Edinburgh, 2019)

The six aspects of Gibb's cycle are listed below:

1. Description of a specific experience of interest
2. Feelings and emotions in a specific situation
3. Evaluation of experience in negative and positive characteristics
4. Analysis to consolidate and summarize in order to make sense
5. Conclusions about the experienced situations
6. Action definition of how to act while confronted with a similar situation

In the following, Figure 2-1, Gibbs' reflective cycle is demonstrated.

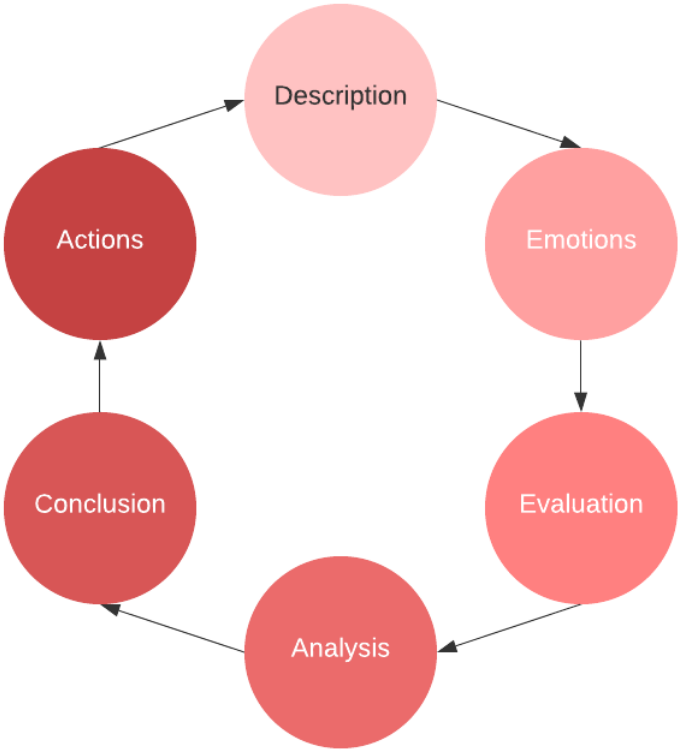


Figure 2-1: Reflective Cycle by Graham Gibbs

Learning from experiences at the workplace by reflection techniques is very useful due to missing learning material, teachers, or learning programs. External motivators can push employees to reflect, learn, and gain knowledge from the past. Typically, employees are kept busy by doing productive activities that are directed to customer needs. During this time, employees often lack explicit time slots to learn actively from sources outside of their working habits. Reflection prompts are especially useful in the workplace as they encourage an employee to reflect on experiences. Very important when it comes to reflection prompts is the right timing of prompts and the context of intervening in the current activities. (Fessl et al., 2017)

2.4 Conversational agents

Conversational agents have been broadly discovered by companies and research institutes, who have developed conversational applications for various use-cases. The implementation of a conversational agents capable of texting and speaking intelligently with humans requires a certain level of technological, architectural and development efforts. Since the steady improvement of artificial intelligence, speech recognition, and natural language processing, the prerequisite for conversational agents has been set. Examples of working applications are bots for speech-based driving navigation, customer service bots for helping customers, or the virtual chatbots acting as a personal assistant for providing weather or traffic information, setting alarm clocks or playing music.

Customer Service bots are effective assistants for businesses. by answering low-level customer queries, they can address customer queries instantly and improve the service quality.

Liao et al.(2016) examined a conversational agent, who was developed to support employees in finding work-related information. The results of their field study was, amongst other things, that agents, sending actively messages, can possibly interrupt users.

Kocielnik et al. (2018) examined chatbots that are supporting companies and organizations for the development of software. They encountered 23 chatbots, developed for Slack, that regard to a daily report of work, project-management or human-resources. One example of reporting working updates is StatusHero, that claims to replace status meetings by collecting progress updates via a chatbot. (Status Hero, 2018) Another example of project management is Nikabot, a bot that collects information of employees on which projects they are currently working. (Nikabot, 2018) The vast amount of these bots, however, are „intended to benefit the worker directly, but instead their team or company“ (Kocielnik et al., 2018).

3. Research Approach

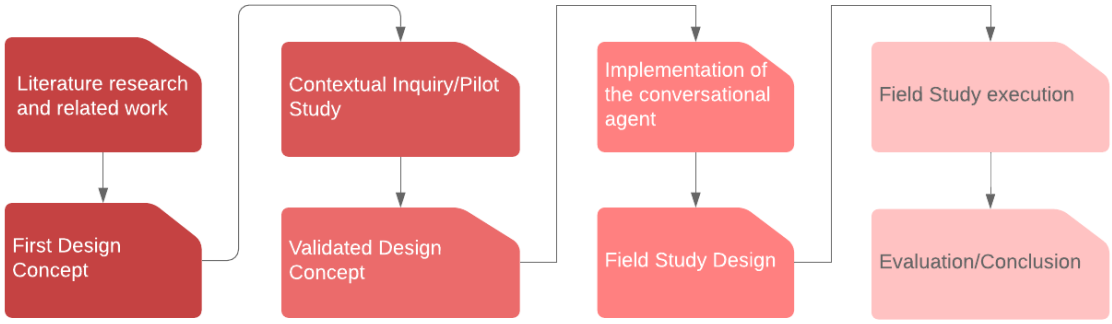


Figure 3-1: Research approach process model

In Figure 3-1, the research procedure is displayed. A literature research with the emphasis on Task & Time Management as well as task fragmentation was conducted. Based on this knowledge, the first design concept was created. This concept included a question & answer conversational agent, which should ask the user: “What do you plan today?”. The user should then be able to respond with two or more activities. At the end of each day, the chatbot should ask the user: “What did you achieve on this day?”.

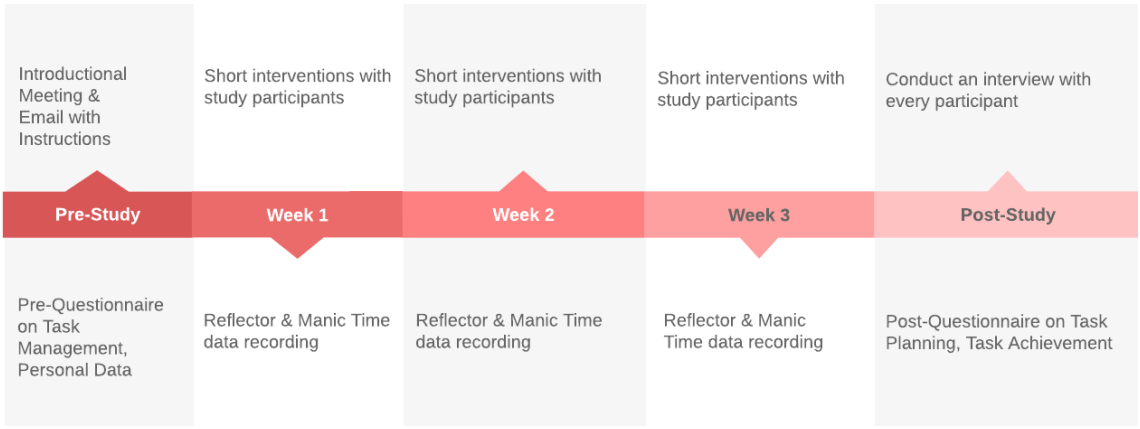


Figure 3-2: Field Study execution overview

In order to validate the design concept, a pilot study, including a contextual inquiry, was conducted. The research was undertaken at the cloud development

department SAP Excellence of All for One Group AG. 5 Employees were interviewed about Task & Time management within their daily working habits. The information and the gained knowledge of the contextual inquiry were aggregated and consolidated with different models, leading to the formulation of a validated design concept.

Reflector, a conversational agent, was developed and enabled a text-based chat modality over Telegram, a natural language processing interface SAP Conversational AI framework and a Java Spring webserver. The interaction of employees with Reflector was examined within a 3-week field study. Ten participants were examined, including a task fragmentation tracking tool called Manic Time.

The procedure of the field study is presented in Figure 3-2.

4. Contextual Inquiry

The design of the realized conversational agent for this master thesis is based on the contextual design approach of Holtzblatt & Beyer (2016). The primary technique of Contextual design is called Contextual inquiry, a method for collecting empirical data. This technique and the realization within a validation study is presented in the following section.

4.1 Method - Contextual Design

Contextual Design is a user-centred design method created by Karen Holtzblatt and Hugh Beyer. The goal of contextual design is to figure out an innovative design based on the idea of collecting data from target groups in the place of interest and draw design implications for a product or a product prototype.

In the early years of software development, product requirement lists, divided into functional and non-functional requirements, were the natural approach to develop a system. Nowadays, technological improvements and the interconnectivity of people create a more sophisticated way of innovation and product conception. The availability of modern technologies anywhere and anyhow, demands a design for life and not just a design for a specific use case and a well-defined environment.

In Contextual Design, a process of multiple actions is proposed. The first level is to conduct a contextual inquiry. After that, interpretation and consolidation steps have to be made. Then, visioning a user environment design and a prototype of a product has to be created.

During a contextual inquiry, a better understanding of a target group doing a specific activity is intended. To achieve this, the researcher should be as close to the target user, watch their activities, and talk to them.

In order to better understand the user, the researcher must collect data about the user's activities. Within the contextual inquiry, an interviewer asks activity-specific questions and displays the data through models. The more interested, curious,

and motivated users will be asked, the better the information will be, the better the foundation for a product can be established.

In the contextual inquiry, a natural human way of interacting is preferred without an exact list of procedures to conduct the interview. Therefore, Holtzblatt & Beyer suggest a master/apprentice model.

The advantages of the master/apprentice model are the “rich detail of everyday life available for observation and discussion. It steers the interviewer and user away from high-level abstract questions. It suggests an attitude of inquiry, attention to detail, and humility. Moreover, it recognizes that the user is the only true expert on their activities.” (Holtzblatt & Beyer, 2016)

In general, software engineers and innovation engineers think about products utterly different as people with no technological background. The mindset of engineers is not equal to the mindset of a person with a completely different background.

The four contextual inquiry principles represent the underlying way of applying the technique in our reality. Focus, Partnership, Context, and Interpretation are vital elements.

4.2 Design Focus

After the design concept was created for a product that builds upon the latest technological standards and considers Task & Time management, interviews with five employees of the target development organisation were conducted.

Each interview has started with a short introduction, in which the idea was explained to each participant. The information collected was guaranteed to be confidential.

After that, the focus of the conversation was altered on the participant’s daily activities & habits, where the participant was asked: “What does your typical day look like? When do you go to work? How do you commute to work? What are your most-used applications during a work-day?

The goal of the contextual inquiry was to encourage the user to talk about his daily life and the activities concerning Task & Time management. The fundamental part of each interview consisted of questions and observations regarding feelings and sentiments concerning task planning, task achievement, and time fragmentation. However, also, the idea of designing for life has been taken into consideration.

Amongst other things, questions in the interview has been:

1. When was the last time you made an important entry in your calendar?
2. Can you show me how you made this entry?
3. Where and when do you plan your activities?
4. Which tools do you use for planning your activities?
5. Are there interruptions during your work?
6. What are the sources of interruptions?
7. What Task & Time management tools do you use?
8. What are the challenges you have using these tools?
9. How effective would you rate these tools for organizing your time?
10. Why did you choose the tools you are using?

4.3 Study Participants

The pilot study was conducted with full & part-time employees within a workplace environment. Five workers participated in a master & apprentice interview about Task & Time management. The participating employees are cloud developers at All For One Group in the department SAP Excellence. The mission of the organization is to develop in-app software extensions for SAP applications. The vision is to help customers become a digitally improved and competitive company.

More precisely, the IT company is a provider for consulting services of SAP By Design, which is a standard ERP application for visualizing business processes and managing business records. Customers of All For One Group are corporations looking for a way to make their value chain more efficient, transparent and intelligible.

The main working activities of the participants are developing customer-specific solutions, which includes conceptual design, implementation, and testing of HTML or JAVA as well as ABSL application. The challenges are achieving tasks within the estimated project scope. The target group is especially faced with the precise estimation of application efforts, which must be determined for contracting a customer order.

Within project-based elaboration of potential improvements in business processes or digital know-how, designing steps, implementing steps, and integrating steps are necessary for the successful hand-over of a solution to a customer.

These solutions contribute to the digital transformation from analogous to digital companies and raise their competitive level.

4.4 Contextual Inquiry Results

The ultimate step in the contextual inquiry was a wrap-up of the interview with an interpretation of the new information about Task & Time management. Besides, the models that have been drawn were shown to the participant.

The comments from the five different participants were formalized and put into three models for structuring and revealing the underlying activities:

- Day in the life
- Collaboration Model
- Affinity Diagram

In the Collaboration model in Figure 4-1, the interconnectivity of people is illustrated. Primarily, it visualizes how the software developers relate activities. A Collaboration model was selected because many stakeholders are involved, and there are more than two communication schemes. A collaboration model provides a better understanding of individuals or groups of individuals interacting in a social system.

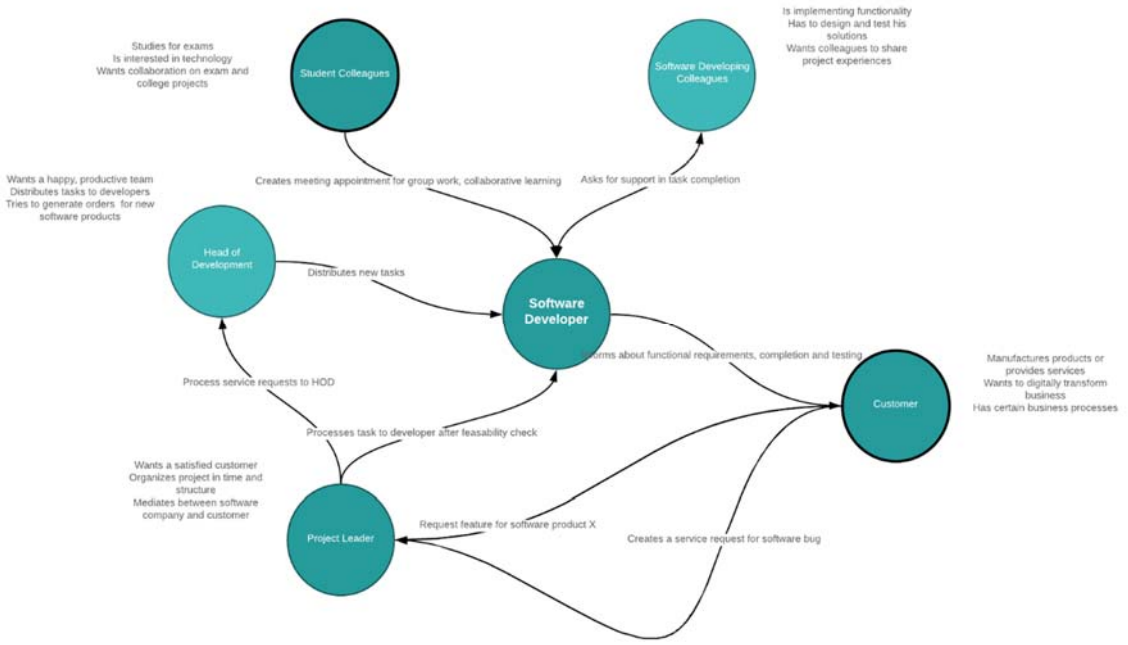


Figure 4-1: Consolidated Collaboration Model

The Day in the life model in Figure 4-2 assists in getting a view on how the target activity fits into the composition of the day. It marks the activities that are performed in different places, times, and platforms.

The model illustrates the overall structure of a typical day and how activities fit into this daily life.

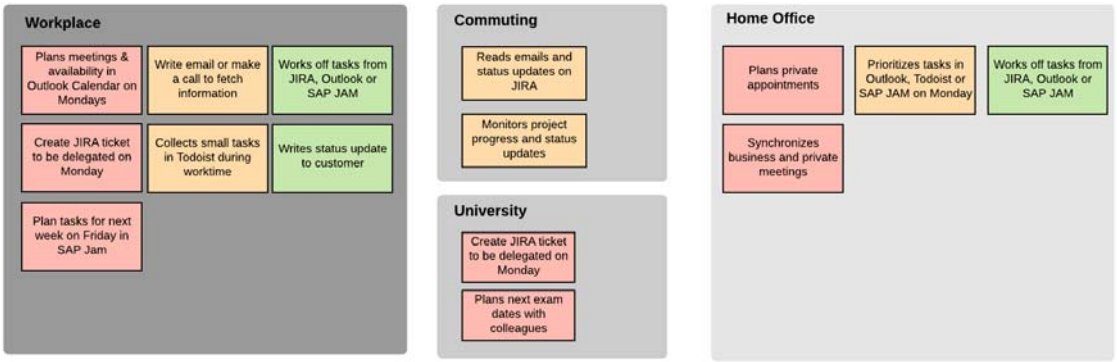


Figure 4-2. Consolidated Day-in-the-life model

The interviewees delivered their activity plans in different periods, more precisely in daily, weekly, and monthly intervals.

A distinctive feature, when analyzing the data, was the point that a large percentage of to-dos emerged from emails in the client inbox folder.

6 out of 10 Participants even reported that they use their email client as a sort of a task managing tool. The incoming emails will get prioritized, categorized, and are removed from the directories as soon as they were done.

Design principles that have been derived from the interviews were that to-dos are mostly temporarily, mobile, editable, small, easy to create, and easy to delete.

4.5 Design Concept

The final design concept emerged from the first concept and the validation within the pilot study.

The first concept was created out of the idea to support Task & Time management by reflecting its own behaviour. By reflecting on these aspects regularly, people can learn about their behaviour of managing tasks and possibly improve it. With a question & answer chatbot, the user is encouraged to think about his task planning and task achievement. The idea was to question: “What do you plan today?”. The user should then be able to respond with one or more activities. Later on, the chatbot should ask the user: “What did you achieve on this day?”.

Aggregating the information collected in the contextual inquiry, the result of the final design concept was a conversational agent with reflecting question & answer functionality. The chatbot should capture sentiment and behaviour.

Based on the contextual inquiry, a conversational agent should be capable of asking questions about Task & Time management before, during, and after work. The conversational agent should be used via smartphone, desktop pc, or tablet. User must be able to input his/her to-do's easily.

Furthermore, the prioritization matrix of Covey should be included. The priority of a task separates the set of tasks in more or lower critical tasks. Covey originally proposed a 2-n matrix, where urgency and importance are the parts of priority. The principles are

1. Differentiate tasks between urgent and important

2. To be effective in time, do the important things at first

3. Very urgent activities must be done immediately, therefore other tasks get delayed

In project management literature, tasks are commonly designed after the goal-setting-theory. According to this principle, tasks should be specific, measurable, achievable, reasonable, and time-bound. A task matches the criteria if the task is unambiguously specified. Every detail must be clear, well-known, not obscure. The ability to measure task completion is necessary for deciding when a task is completed or not. A task should be aspirational and reasonable. It is not beneficial to stick to unachievable tasks, and a fixed end should terminate a task.

Further functional and non-functional requirements were a user interface, allowing text or speech-based input. Additionally, the agent must be able to collect, comprehend, and answer to the messages from a user. Moreover, the agent must store the collected input in order to retrieve the obtained information. The chatbot's response times should be appropriate for a decent conversation. The chat input from users should be available for analyzation purposes. The chatbot should be usable with smartphones, tablets, or desktop pc. The developing process should be manageable regarding the total effort for a single developer.

Furthermore, a Task & Time management tool should be: Instantly on, to support quick and easy input and clear visualization. (Bellotti, et al., 2004). The description of goals typically starts a classic Task & Time management process. The goal then is divided into smaller chunks, tasks that logically derive from the goal.

5. Implementation of a Conversational Agent

The initial design conclusions and the results from the contextual inquiry were merged to a final design concept for a conversational agent concerning Task & Time management. The chatbot will be presented in this chapter.

5.1 Interaction with Reflector

Reflector is a conversational agent designed to support reflective behaviour on Task & Time management. The interaction with Reflector is made possible by Telegram, an instant messenger on mobile phones. Users can send and receive messages from the chatbot. The chatbot can answer, query, and remind users with message prompts to reflect on Task & Time management. Moreover, Reflector can be used as a Todo-List Manager that adds, saves, assigns, and deletes to-dos. Telegram as the instant messaging app allows the user to react on notifications from Reflector, reminding for half-day reflection cycles. A user can then answer questions regarding daily activities in their life, rethinking experiences or situations daily. Reflector reacts flexibly to various emotional states as well as positive or negative Task & Time management habits. Reflector intends to motivate users to deal with Task & Time management challenges.

5.2 Conversational Structure

In this section, Reflector's dialogue structure is discussed.

The conversational structure of Reflector is visualized in Figure 5-1, Figure 5-2 and Figure 5-3, which are complementary parts of the entire dialogue structure. Each conversation between the chatbot and the user is constructed of an array of elements. A coloured rectangle represents a conversational element. The directed paths between the objects are ties to the upcoming conversational element. Generally, the conversation flow is drawn from top to the bottom elements, where the first elements are greeting messages.

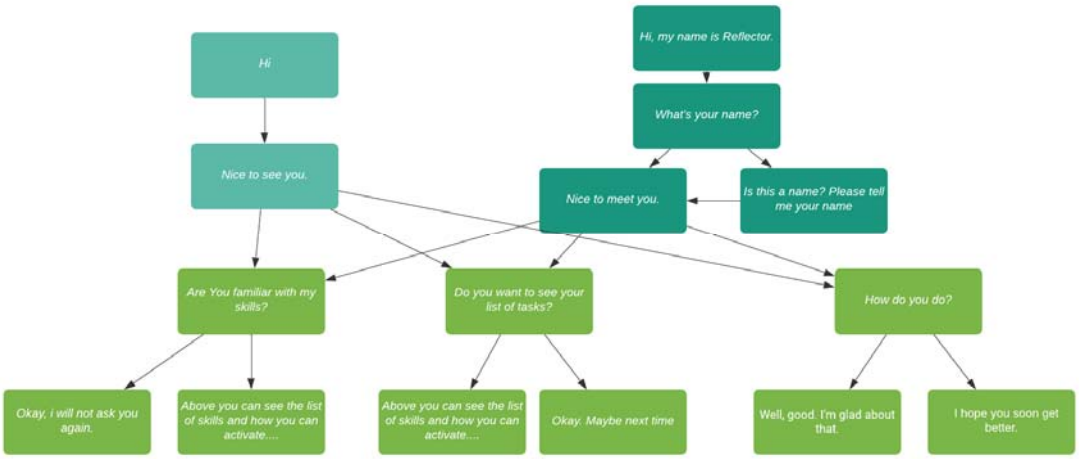


Figure 5-1: Greeting & Introduction conversational elements

Turquoise objects on the left flow do represent the login step, which is typically a plain "Hi" message followed by the message "Nice to see you". The darker-coloured Turquoise objects on the right side are displayed when a user uses Reflector for the first time . The message emitted from the chatbot acts as an introduction message, saying "Hi, my name is Reflector". After the initial sentence, the chatbot asks for the user's name: "What is your name?". If the user types a name that is recognized by the entity detection, which is a validation of the user's input, the message flow transits to the next element, "Nice to meet you". If the user types a name that is not an actual name, the conversation state is not changed, the chatbot asks the user for his name again with the question: "Is this a name? Please tell me your name."

The green-coloured messages indicate the second layer of conversation. After the greeting of users, the chatbot tries to give the conversation a road. Therefore, three following ways have been implemented. Reflector tries to support the user's knowledge about his skills. That is why Reflector asks on the left message flow if the user is familiar with the skillset ("Are you familiar with my skills?"). The answer is either the list of all implemented skill features, that will be explained in the next section in more detail. Alternatively, the answer, "Okay, I will not ask you again.". If the user answers with yes, this introduction question will not be asked again, as Reflector assumes that the user does not need any explanatory on Reflector's

functionality. The technical implementation of this behaviour is explained in chapter Spring Webserver.

The right and centred message flows are alternating, changing every day. The reason is to avoid monotonous dialogues and boredom of users chatting with Reflector. The central message flow asks the user: "Do you want to see your to-do list?". The chatbot tries actively interacting with the user and recommends an overview of the list of tasks. The right message flow asks the user about his/her sentiment: "How do you do?" If the user enters a negative statement, the chatbot tries to cheer up the user. If the user enters a positive sentiment and it gets detected, Reflector answers with an acknowledging answer.

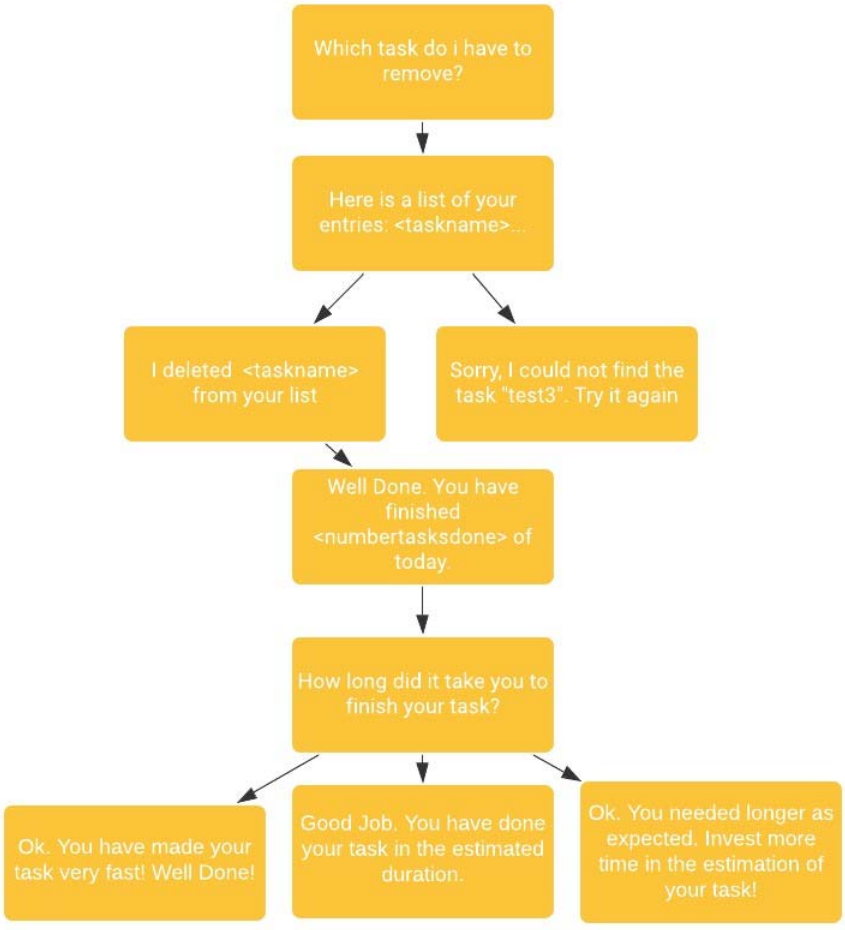


Figure 5-2: Conversational structure for removing a task

In Figure 5-3 the reflection dialogue structure of Reflector is illustrated.

The left-side message flow coloured in red represents the Questions & Answer feature. The orange-coloured message flows are referring to the task list features of Reflector. The feature set includes adding new tasks, removing existing tasks, and assigning tasks to other colleagues.

Figure 5-3: Conversational elements for reflecting on Task & Time management

The crucial functionality of Reflector is the Question and Answer feature. A user can open Telegram, navigate to Reflector's chat, and start a reflection. Then, Reflector asks the user several questions regarding the personal usage of time. Each of the questions is written in English and is part of a set of questions. The chatbot provides different sets of questions depending on the time of the day and week. The questions selected for each set origin from different subject pools, where each pool is labelled with a category regarding task planning till task interruption.

A subject pool includes classical Task & Time management topics. The subject pools defined for the question sets were

1. Short-term & long-term task achievement,
2. Sentiment & emotion
3. Short-term & Long-term planning
4. Interruptions and task fragmentation.

In the following table, the questions for Monday morning are listed. Five questions are questions treating interruptions. One question is about task planning. During the process of finding reasonable questions, the fact that repeating questions would quickly decrease the motivation has been taken into consideration. That is why questions repeat itself max. three times.

Furthermore, the questions in the morning are entirely distinct from those in the afternoon. Usually, questions regarding task planning were asked in the morning. Questions about task achievement were asked in the afternoon. Since the field study was designed for a period lasting three weeks, on Monday's task planning on the whole week was sent. Every Friday, the chatbot sent task reflecting questions for the previous week. The questions and the sequence of questions were inspired by the reflection cycle of Gibbs.

Table 1: Reflective questions of Reflector regarding planning and achieving

Reflective Questions
What tasks are you planning for today?
How did you feel last time, when you achieved a task from your task list?
What were some of the most satisfying moments at work for you yesterday?"
What have you learned from the activities yesterday?
Could you describe your strategy for approaching this unpleasant task?
Will you do anything differently if you do this task, or a similar task, in the future?

Table 2: Reflective questions of Reflector regarding fragmentation

Reflective Questions
When and where was the last time something interrupted you doing a piece of activity?
Was the interruption from a source within the same context of your original work?
How urgent was the interruption?
Did you complete the task that was interrupted till the end of the day?
Have you planned times to allow interruptions purposely?
What tasks are you planning for week?

5.3 Application Architecture

The overall technical environment is shown in Figure 5-4. The sequence of the interconnected messages between the different components is visualized in Figure 5-5.

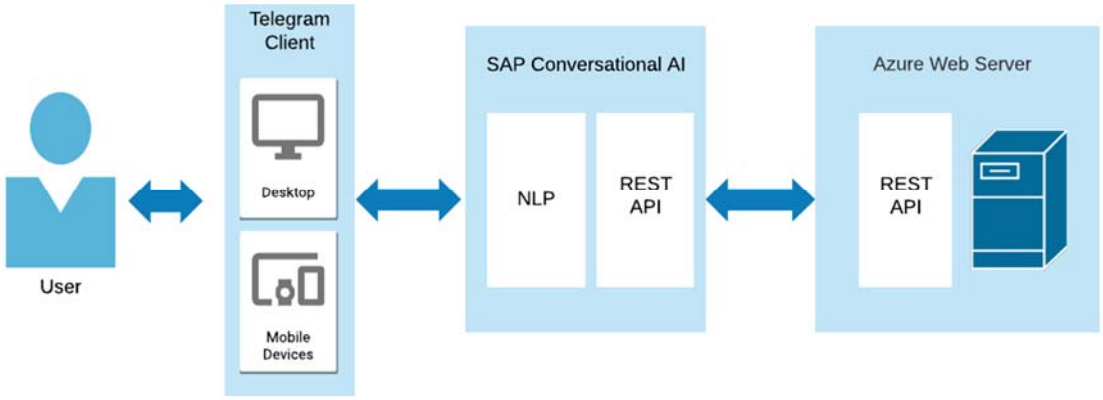


Figure 5-4: Web architecture of Conversational AI, Webserver and Client

SAP Conversational AI has been chosen as a platform for implementing a conversational agent. The framework provides a simple UI for designing conversation paths. It allows for simple versioning and multiple environments. Additionally, the Conversational AI has NLP technology for considerable speech processing. Finally, the analyzation functionality of SAPCAI allows investigating usage and bot statistics over a dashboard.

The choice for the backend component communicating with SAPCAI was a JAVA Spring Boot web application. Java is one of the most established object-oriented programming languages. It provides features such as constructors, interfaces, in-memory management.

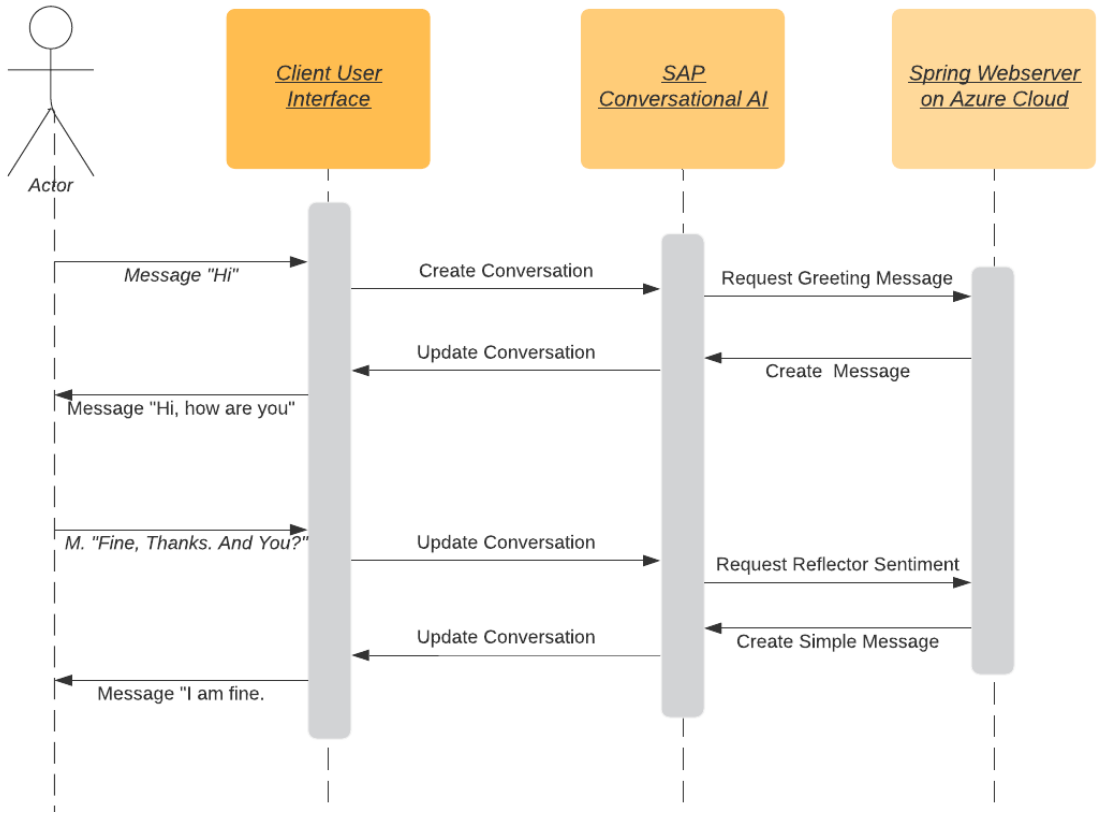


Figure 5-5: UML Sequence of Reflector's Greeting Process

Telegram represents the User Interface (UI) of Reflector. The user interacts with the Telegram chat window that pipelines messages to the Bot Connector. The Bot Connector then passes the input to the Bot Builder, who makes a webhook to the Java backend server. Different inbound and outbound communication scenarios are implemented to allow more flexible and extensive chatbot functionality.

5.4 Client-User-Interface

The User Interface for interacting with Reflector is handled by Telegram, a messaging app with a focus on security. The application is free and is supported by different devices, including smartphones, tablets, or computers.

Telegram is a messaging tool, founded by Pavel and Nikolai Durov in 2013, that enables users to send and receive messages, photos, and other media types. Telegram has a chatbot API for managing bots in private and public chat groups. The set of Representational State Transfer Application Programming Interface methods enables the integration of chatbots.

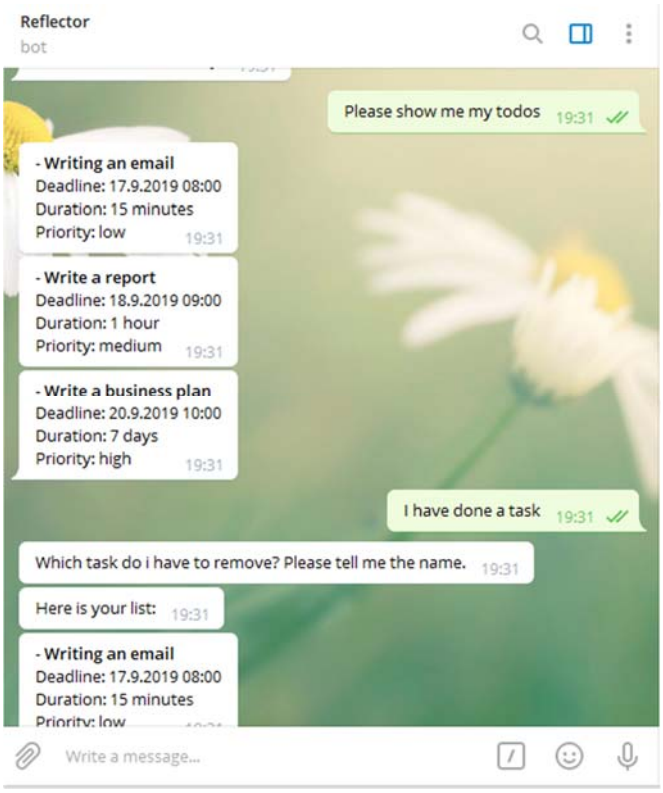


Figure 5-6: Reflector conversation for removing a task in Telegram Messenger

Figure 5-6 shows Telegram's chat window and Reflector as a conversationist. In the chat history, the chat functionality "Task Management" is illustrated. It shows the process of a user striving to delete a task from his/her list of tasks. By typing "I have done a task," the chatbot is alerted. The response is: "Here is your list:" enumerating every task that has been saved by the user. The user then writes

the task description “Writing an email” to specify the task for deletion. The chatbot confirms: “I deleted ’Writing an email’ from your list”, “Well Done. You have finished the next task for today!”, “How long did it take You to complete the task?” The user responds: “15 minutes”. The chatbot applauds to the achieved task: “Good Job. You have done your task in the estimated duration”.

5.5 SAP Conversational AI

SAP Conversational AI is a framework to develop, train, build, connect, and analyze chatbots. Engineers are provided with instruments to develop chatbots supported by natural language processing. The workbench of the Conversational AI is capable of versioning and setting up different environments for organizational accounts as well as multiple languages. The chatbot framework is integrable with SAP and non-SAP solutions and carries text-based input as well as speech-based input.

Within the Conversational AI framework, the implementation of a chatbot is structured in 3 implementation actions. The first action is a training step, in which intents are declared. The second step includes modelling skills, and conversation flows within a workbench, which can be seen in Figure 5-7 .The third step is the monitoring area for analyzing the user input.

For training a chatbot, one must create intents and enrich those with expressions. Basically, intents are defined as a list of expressions for identifying one specific, wanted purpose. The more expressions, the better the intent can be categorized. The intents in the entire dialogue structure must be well separated and represent a distinct quantity. Expressions are understandable sentences, words, or phrases. For multilingual support, each expression must be translated into each language. The SAPCAI framework has the solution of uploading a set of expressions with a conventional excel spreadsheet.

In order to give each word meaning, entities must be declared in the training step. Entities are keywords that can be detected within an expression. By defining entities, the framework can automatically label words or phrases within expressions. The framework has 28 prebuild entities such as date times, locations, or person

names. With increasing feedback given, the bot can improve the accuracy of comprehension.

Furthermore, entities are categorized in Free, Custom, and Predefined “Gold” entities. Gold entities cannot be trained as they are detected automatically. The list of gold entities consists of Date Time, distance, duration, speed, numbers and several more. Additionally, so-called gold-entities are enriched with data. E.g., the word tomorrow gets detected and enriched with a formatted Date Time. Custom entities are separated into independent and restricted entities. This provides additional possibilities for intent detection. Independent entities are used, when the set of expressions is not known, and machine learning should enrich the entity with new expressions.

The training of independent entities is achieved by either tagging words in a sentence or adding synonyms. A restricted custom entity is built up by a fixed list of expressions, meaning that every word must be defined. Otherwise, the intent is not activated.

Sentiment detection is an integral part of analyzing a user’s input. We decided to follow guidelines suggesting a higher granularity of sentiments. This allows for treating different levels of positive and negative inputs.

The conversational framework also understands 4 acts of sentences, yn-query, wh-query, command, and assert.

To build a chatbot with the builder workbench, one must define skills. A skill requires a trigger, which are conditions that must be met. Conditions in SAPCAI can be intents or entities, as well as positive or negative detected sentiments. Each condition can be combined with or-conjunction / and conjunction. As soon as a condition is met, requirements can come into place. A requirement in SAPCAI is defined as a piece of information that is explicitly required and stored in memory. Without meeting the requirements, the skill is not further executed. A requirement can be entities or intents.

Finally, when triggers and requirements are fulfilled, the action of a specific skill is executed. Actions are typically messages to the user, a webhook or API call, fallbacks, memory update, change of language, or the execution of another skill.

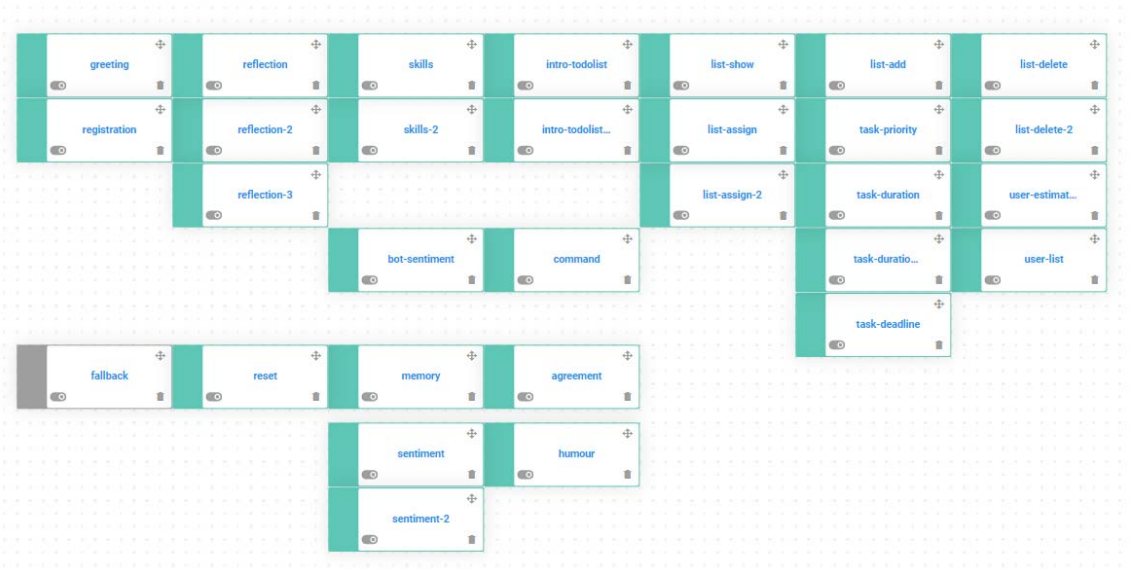


Figure 5-7:SAP Conversational AI Builder Workbench with Skills

In the illustration above, the workbench for modelling the communication flow is visualized. Every skill is represented by a turquoise rectangle consisting of triggers, requirements, and actions. Altogether, Reflector consists of 30 skills, where the turquoise skills are typical skills, and the grey skill is the fallback skill if none of the skills is triggered. The skills are clustered in the context of purpose but do not necessarily depend on each other. There is a link between the rectangles visible. However, the actions defined within a skill could refer to any of the skills.

5.6 Spring Webserver

Reflector has three relevant components to enable a meaningful conversation. Beyond a user-interface and a natural language processing unit, a JAVA Spring Webserver was developed for operating the logical control instance. Every mes-

sage is recognized and categorized by the Conversational AI framework and directed to the controlling server logic subsequently. The server receives, processes, and responds to the requests to the previous layer.

The system backend is coded in the object-oriented programming language Java. The source code is built, assembled, and compiled to an executable, hosted on Microsoft Azure. Azure acts as a platform for the operating webserver. A Linux runtime environment provides the server functionality at https://caiserver.azurewebsites.net.

Every message containing valuable data for processing in detail is sent from the SAP Conversational AI to the REST interface via webhooks. The server implements methods for handling the incoming REST-full requests, processing the information sent, and responding in the predefined API specifications of SAP Conversational AI. A respond message body must be structured precisely according to the specifications of the Conversational AI.

The class “RESTInterface.java” takes requests that are incoming on https://caiserver.azurewebsites.net and processes the data accordingly. The requests in this context are standard Hypertext Transfer Protocol requests with header and body information.

The header Http-method property must be set to either POST, GET, and PUT. The payload is formatted in JavaScript Object Notation and includes message-based, natural-language, and meta information.

To simplify the source code readability, Jackson annotations convert the payload data to Java objects. As soon as a REST API function is called, the properties are serialized and passed as parameters to a corresponding class constructor, which matches types in the payload. By constructing instances with Jackson annotations, the code is clean, performant, and the processing of the information is done efficiently.

In Figure 5-7, extraction of the REST API source code is presented.

```
@PostMapping("/question/{id}")
public ResponseEntity<String> getQuestion(@PathVariable("id") Integer id,
@RequestBody Text text) throws IOException, ParseException {

    String result = MessageHandler.getInstance().getQuestion(id, text);
    return new ResponseEntity<>(result,HttpStatus.OK);
}

@PostMapping("/answer/{id}")
public ResponseEntity<String> setAnswer(@PathVariable("id") Integer id, @Re-
questBody Reflection reflection) throws IOException, ParseException {

    String result = MessageHandler.getInstance().setAnswer(id, reflection);
    return new ResponseEntity<>(result,HttpStatus.OK);
}
@GetMapping("/reflections/{id}")
public ResponseEntity<String> getReflection(@PathVariable("id") Integer id)
throws IOException {

    String result = MessageHandler.getInstance().getReflection(id);
    return new ResponseEntity<>(result,HttpStatus.OK);
}

@GetMapping(value = "/low.png", produces = MediaType.IMAGE_JPEG_VALUE)
public @ResponseBody byte[] getLowPrioIcon() throws IOException {
    InputStream in = getClass().getResourceAsStream("/icons/low.png");
    return IOUtils.toByteArray(in);
}
```

Figure 5-8: REST API of Reflector

The platform for deployment is Microsoft Azure, a cloud computing service for creating, testing, and maintaining applications. (Microsoft Corporation, 2019). The webserver is running via an app service built on a Java stack. The runtime version is Tomcat 8.5. Hypertext Transfer Protocol packages handle the communication setup, secured by Transport Layer Security 1.2. The server's Random-Access memory is configured with 1.75 Gigabyte working capacity. The continual in-memory space is configured to 10 Gigabyte. The virtual IP address is 13.74.252.44. The webserver runs with the Microsoft Azure version 2.1.6.

Apache Maven is used for the project build cycle as it simplifies the process, it enables a homogenous build system, and it allows migration to new features. Maven is responsible for dependency management and automatic updating of the dependencies. The project folder build configuration file is of type POM (Project Object Model), which includes the build configuration for the project. The POM file is configured with the packaging definition “war”. The java version for the building is set to 1.8.

The most important dependencies are GSON, OpenCSV, Azure deployment, and Spring initializing dependencies. Gson is an open-source Java library to serialize Java objects to JSON and deserialize JSON objects to Java objects. OpenCSV is used for parsing the comma-separated files. The question pool for Reflector’s reflection questions is parsed via CSV files. The com.microsoft.azure dependency is used for continuous deployme1nt of the server to the deployment platform.

Spring has been chosen to embed tomcat as a runtime environment. Spring is basically an open-source application framework. In this project, Spring was selected for providing functionality with web services and the easy to implement REST API. Spring Boot has been for initializing the project directory and the overall project structure.

6. Field Study on Work-based Reflection

This chapter discusses the research questions and results of the field study. Additionally, used tools and question forms are presented. The study context and the participants and the procedure of the study is shown.

6.1 Research Questions

The objective of this master thesis is to investigate the Task & Time management behaviour of software engineers and discuss the results and observations of the study participants. Furthermore, the task fragmentation of software engineers and the interaction with a conversational agent is the focus of attention. The research questions that derived from the continuing work on the objective were:

Research question 1a: What tools and which behaviours of Task & Time Management can be examined in the target user group?

Research question 1b: How fragmented is the daily work of the target user group?

Research question 2: How extensive has "Reflector" been used, and how helpful did the target user group find the chatbot?

6.2 Tools (Questionnaires, Manic Time, Chatbot)

Manic Time did record various kinds of information regarding application usage, application fragmentation, and worktime resources.

Furthermore, metadata about the current application, the status of activeness, and every used document on their desktop pc was recorded. The data included exact timestamps of working activities. Accumulated values, such as working times, the total usage time of each application, and additional cumulative values have been collected.

The chatbot intended to encourage participants to answer questions regarding different reflection-based topics such as short-term or long-term planning, short-term or long-term task achievement and task fragmentation.

The Pre-Questionnaire at the beginning of the study was sent to each participant via email. The questionnaire contained 16 questions in total. The answered questionnaire was submitted with the participant code, which had to be constructed from each participant at the questionnaire. Every file or data aggregation was anonymized with the created participant code.

A participant code consisted of 5 alphanumerical characters:

1.The first letter of the place of birth.

2. The day of birth (two digits)

3. The first letter of the participant's father's first name

4.The first letter of the participant's mother's first name.

6.3 Study Context and Participants

The field study was conducted with part & full-time employees in a workplace environment. Ten workers participated in a 3-week field study about Task & Time management. The participating employees are developing in-app software extensions for cloud applications as well as side-by-side applications for SAP products.

All the study participants are employed at the All For One Group, working in Cloud ERP or Customer Experience departments. The IT company is a provider for consulting services of SAP products. The customers of large and middle-sized corporations are supported in their digital formation process.

The working area of the participants is the development of customer-specific solutions, which includes conceptual design, implementation, and testing. The target group was confronted with problems of effective Task & Time management and efficiently achieving tasks within projects. The target group is additionally confronted with the approximate estimation of personal workload and the assessment of work payloads. Due to minimal resources, the employees are often working on multiple customer projects. Thus, they have a continuing necessity for planning, implementing, testing, and monitoring the project progress.

Within project-based elaboration of potential improvements in business processes or digital know-how, designing steps, implementing steps, and integrating steps are necessary for the successful hand-over of a solution to a customer. These solutions contribute to the digital transformation from analogous to digital companies and raise their competitive level.

The educational level of the participants is University, High school, and Grammar school level. Seven employees claim university graduation as the highest educational level.

6 participants were male, 4 participants were female gender. In Figure 6-1 , the age of the study participants is shown. 3 participants were between the age of 20 and 23. 4 participants were at the age of 24 till 27 and 3 participants were 28 or older. In Figure 6-2, the employment of every study participant is illustrated.

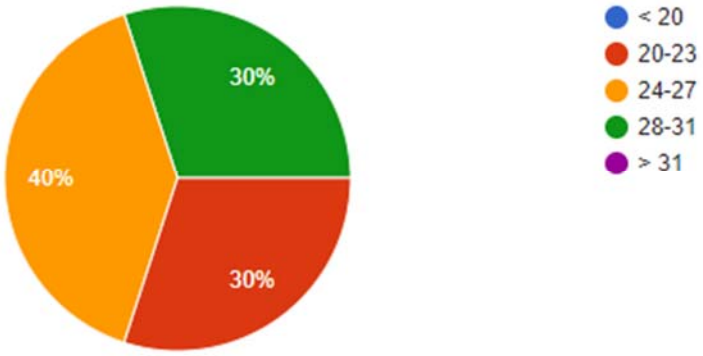


Figure 6-1:Age of study participants

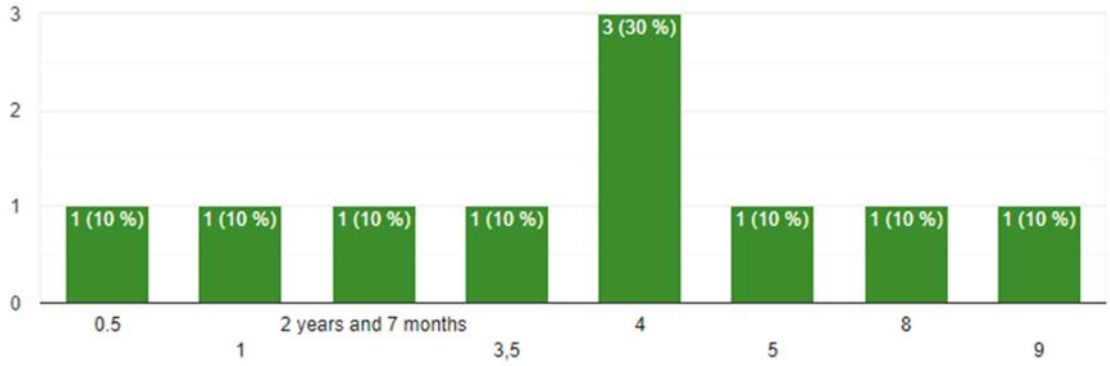


Figure 6-2: Employment of participants in Years

6.4 Study Procedure

At the beginning of the field study, every participant had to answer a questionnaire containing personal, reflection-based, and chatbot-based questions. Then, every study participant was instructed to record their working activities by a time tracking tool for three weeks. The recording took place at the workplace from

Monday to Friday. The idea was to use the time tracking software over a specified period to collect data about their application usage & fragmentation.

Additionally, participants were encouraged to use Reflector, the conversational agent for reflecting tasks & time resources.

6.5 Data Analysis

Before the field study started, each participant had to fill out the digital Pre-Questionnaire. Manic Time recorded the participant's computer activities. Reflector, the chatbot, gathered data through the reflection feature, and ultimately, an interview with participants was held.

At the start, each participant was instructed to the basic functionality, the anonymization of data, and the usage of Manic Time. After three weeks, information details about their application usage were collected. The analyzation of Manic Time data points required the export and collection of the trace files generated on each of the participant's desktop devices.

The data displayed in each export file was examined with respect to the duration each application was in charge. After that, the minimum, the maximum and the average usage time was calculated,

The results are visualized in different charts and descriptive text parts. The statistics cover how the bot is used, the number of conversations persevered with the bot and at what time the users typically interacted. Messages between the bot and the user, conversation patterns over the week as well as skill and entity scores are included.

7. Results

Results of the field study regarding Task management, fragmentation and reflection conducted from 12th to 31st of August, are listed in the following sections.

7.1 Task & Time Management – Tools & Behaviour

In the following section, results of questionnaires, interviews and content from chatbot conversations are presented.

The tools of participants, planning, maintaining and monitoring tasks were extracted from Pre-, Post-Questionnaire and interviews. Regularly, to-dos are collected with different tools and different devices. Tools that were used for collecting tasks have been JIRA, JAM, Outlook, Gmail, Microsoft Teams, Slack, Todoist, Excel, Google Calendar, Evernote or pencil & paper. Pieces of information were names, telephone numbers, email addresses, URLs, references, blog entries, passwords, code snippets, meetings, messages, specifications, directory paths. JIRA has been used most often mentioned when talking about Task & Time management tools. Outlook was used as calendar to coordinate the dimension of time, whereas the email inbox is a collection of tasks with different priorities. After that, classic pieces of paper, such as post-its or to-do managers, as well as electronic or paper notes, are used. Very few participants stated they use Excel or Google Tasks or no tool at all.

Results from Pre-questionnaire showed that task planning is a very internalized behaviour in the investigated research environment. Task planning represents a very significant part of the overall Task & Time management behaviour. Every participant claimed to write a weekly list for his/her to-dos weekly, daily tasks are derived from that. One participant stated to write a monthly list of goals to achieve. Another participant mentioned that he/she writes and maintains a list with a time span of 5-years or more. Two participants reported that at the end of every working week, a plan with activities and appointments for the next week is made. The participants created a week plan typically and then split their tasks into daily work packages. These working packages are considered, when fixing other appointments, “Look at my fixed/scheduled events (e.g. customer meetings or private

dates) and then build my flexible activities around it." Activities that cannot be planned around scheduled meeting activities, would be procrastinated.

During the Post-Questionnaire, participants were asked what long-term goals they have set for themselves. Long-term goals were defined as goals with at least a 5-year range for achievement. One of the participants said: "Be an expert in the field I am interested in, continuous learning". In total, three participants said they want to become subject matter expert in developing cloud applications. 2 participants said, they want to be happy with their daily work and "always being passionate about what I am doing". 2 participants said, they want to become more independent, financially and regarding their employment. One participant said he wants to get more responsibilities. Two participants said, did not respond with goals or did not have thought of. Another long-term goal was to finish their curriculum or manage to buy a condominium.

Furthermore, results from the Pre-Questionnaire showed that 5 out of 10 people prioritize their daily task list within a tool and do not write it down on paper. After the participants collected all tasks from different tools or platforms, they prioritized the previously collected tasks. "I read my email inbox, collect calendar entries and tasks, and write it down. After that, i prioritize and make my plan for the week." All the participants claim to have one or more priorization criteria, where the most priorization property is either urgency or importance or both.

One participant said: I prioritize by how time sensitive the customer's issue is and how much the business impact grows by the amount of time passing.

The answers to priorization were also "I use the e-mail inbox as task list and categorize by importance of customer", "important things first (or second); using time gaps effectively", "Most important = project with nearest deadline", "Business Impact & time priority", "urgency, dependencies, interest", "I prioritize work by figuring out an order of necessity by time, meaning I focus on the tasks first that are needed to proceed to the next task related to the first."

In Figure 7-1 the participants were asked to rate their agreement on the expression: "I think it is important to have a good Task & Time management"

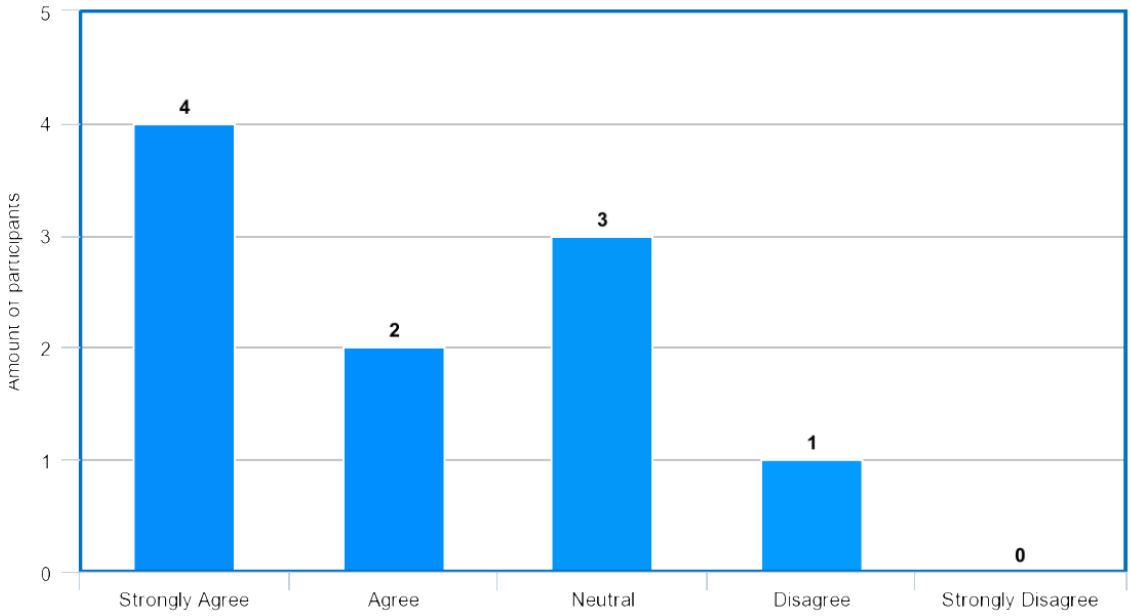


Figure 7-1: Personal perception of Task & Time management importance

In the Pre-Questionnaire of the study, participants were asked on a 5 point-Likert-scale from Strongly Agree to Strongly Disagree, what they think about the question: “We as a project team, often reflect on our work to improve it”. Three participants strongly agreed. Six participants did face this question neutrally, one participant disagreed.

7.2 Task Fragmentation – Application & Strategies

Every participant was instructed to install, track, and review Manic Time statistics. The following table illustrates the application usage patterns of each participant. Each row consists of the participation code, the total amount of data points provided, the average application usage in seconds, the maximum usage time in seconds, and the application usage median in seconds.

Table 3: Overview of calculated participant's application usage times

PCode	Data points(n)	Average usage(h:mm:ss)	Max. usage time (h:mm:ss)	Median(h:mm:ss)
G2TK	115900	0:00:24	0:35:32	0:00:10
D08BM	110000	0:00:25	0:49:25	0:00:10
L24RE	48000	0:00:28	0:25:23	0:00:11
S24IP	44000	0:00:44	0:26:41	0:00:12
G01JC	114300	0:00:21	0:23:52	0:00:10
G31KD	50500	0:00:28	0:47:35	0:00:13
G16KK	35000	0:00:42	0:32:27	0:00:12
G05RI	39000	0:00:37	0:25:16	0:00:14
L19RM	57800	0:00:30	0:55:12	0:00:15
B16RM	55100	0:00:24	0:33:07	0:00:13

Consolidating the data points, the shortest average application usage time was 21 seconds from participant G01JC. The longest average application usage time was 44 seconds of participant S24IP.

The shortest median value of application usage time was 10 seconds from participants G01JC, D08BM, G2TK. The longest median value of average application usage time was 15 seconds of participant L19RM.

Reflector asked participants about their task fragmentation. A question asked was their subjective view on interruptions: “What do you think about interruptions in general?”

The responses showed that the participants were aware of interruptions and that have strategies against it partly. One participant said about interruptions, that

“They are somehow inevitable, but i think i should reduce the amount of interruptions.” One participant stated, that only after repeated interruptions he starts trying to shut down the source by deferring the requests or avoid it somehow. One participant stated: I am not concerned on interruptions. I make it clear when I need my space. Another question Reflector asked the participants was: “What strategies do you have in order to avoid or minimize interruptions during a workday?”

Many participants said that an excellent way to reduce disturbances is to “disable social media (skype, Microsoft teams) and other communication platforms”. One participant said: “create appointments in my calendar with only me participating so that others can see that I am blocked and not available”. One participant said that preparation decreases personal interruptions. One participant stated that the Pomodoro technique is helpful, saying: “...I have problems with staying focused for longer periods, thus I usually work 15 min focused, then take a small break and then repeat the process until work is done.”

Half of the participants stated that turning off notifications purposely on several devices reduces the number of interruptions in their point of view. Nine out of ten participants said that working from home minimizes the level of interruption significantly. Four of ten participants block their calendar in order to avoid Skype callings from colleagues.

In the post questionnaire, participants were interviewed and asked about their opinion, if a tool like Manic time helps them in their awareness of task tracking and fragmentation. Half of all participants said that Manic Time is useful for their time booking. One participant said that “Manic Time is a nice tool because it simplifies my time recordings and I can look up my activities, whenever I want”. Three participants said that the statistics provided are interesting, because of the visually displayed fragmentation. A visualization of Manic Time can be seen in Figure 7-2. The green bar at the top of the illustration represents the user’s status being active/inactive. The bar in the middle represents the application that is used at the point in time written above the active/inactive bar. The fragmented bar at the bottom represents the documents during the application usage.

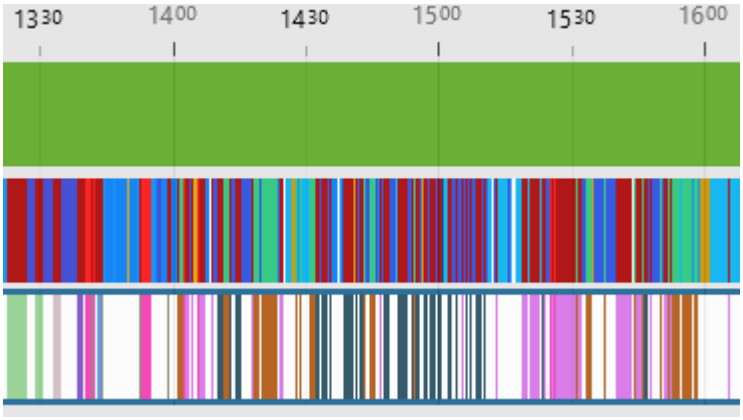


Figure 7-2: Manic Time visualization of participant's working hours between 13:30 and 16:00

7.3 Conversational Agent – Usage & Acceptance

The results of this section originated from the final interview, the Conversational AI framework and the Post-Questionnaire. Based on these data sources, the following questions of the conversational agent can be answered:

1) How often did the participants used Reflector?

2) How did the participants perceive Reflector and how helpful is Reflector for Task & Time management?

The next section starts with answering the question:

1.) How did the participants used Reflector?

Reflector has been used primarily via the instant messenger Telegram on smartphones. In Figure 7-3, the conversations between the participants and the chatbot are illustrated. The usage metric is based on the number of conversations every day for three weeks starting with the 12th till the 1st of September. The maximum conversations during the field study on a single day were 30 conversations. The valleys plotted can be traced back to work-free days.

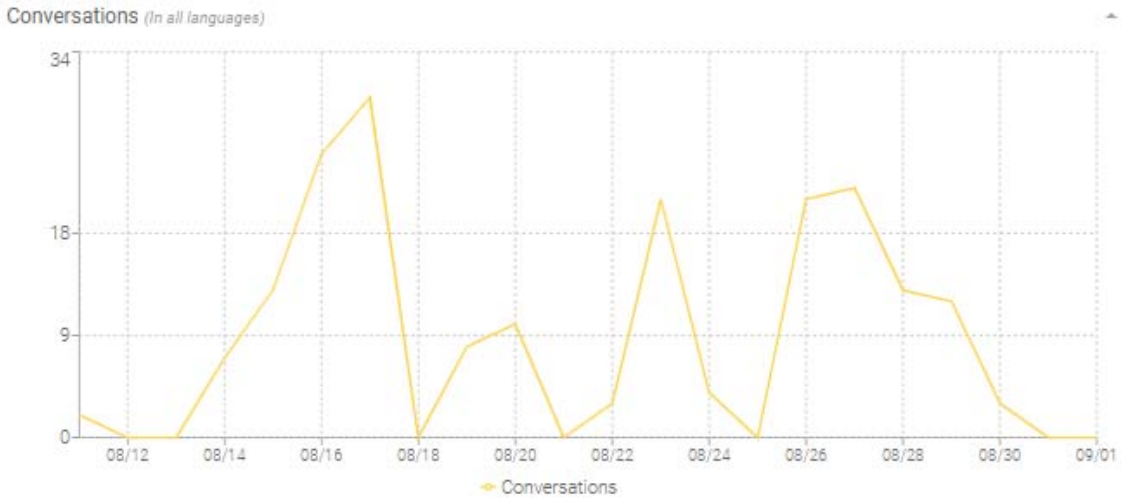


Figure 7-3: Conversations with Reflector during the field study

Figure 7-4: Average messages by conversation during the field study

In Figure 7-4, the average messages sent per conversation is shown. The beginning of the recording is marked on the 12th of August till the 1st of September. The maximum conversations on one day took place on the 23rd of August, with approximately 21 messages per conversation. On the 27th of August, the average message number was at approximately 20 messages.

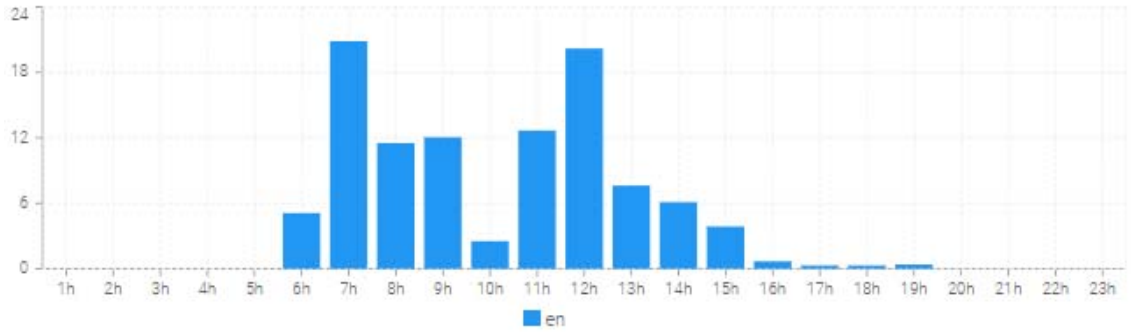


Figure 7-5: Average messages/conversations during day

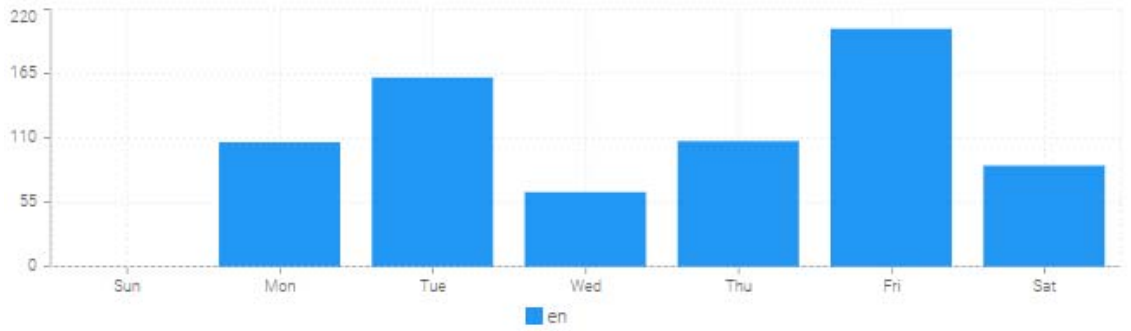


Figure 7-6: Average messages within a week

Figure 7-5 shows the total number of messages received during the daytime. The participants began to chat with Reflector, starting at 6 AM. The latest messages were sent typically at 7 PM. Peaks of message sending can be identified around 7 AM and 12 AM.

Figure 7-6 illustrates the average total incoming messages per week. On Friday, most of the messages were sent to Reflector. The chatbot received 152 messages in total. Second, most messages were sent on Tuesday with 121 messages, followed by Thursday and Monday with both 80 messages. The fewest messages were sent on Wednesday with 48 messages and Saturday with 65

messages.

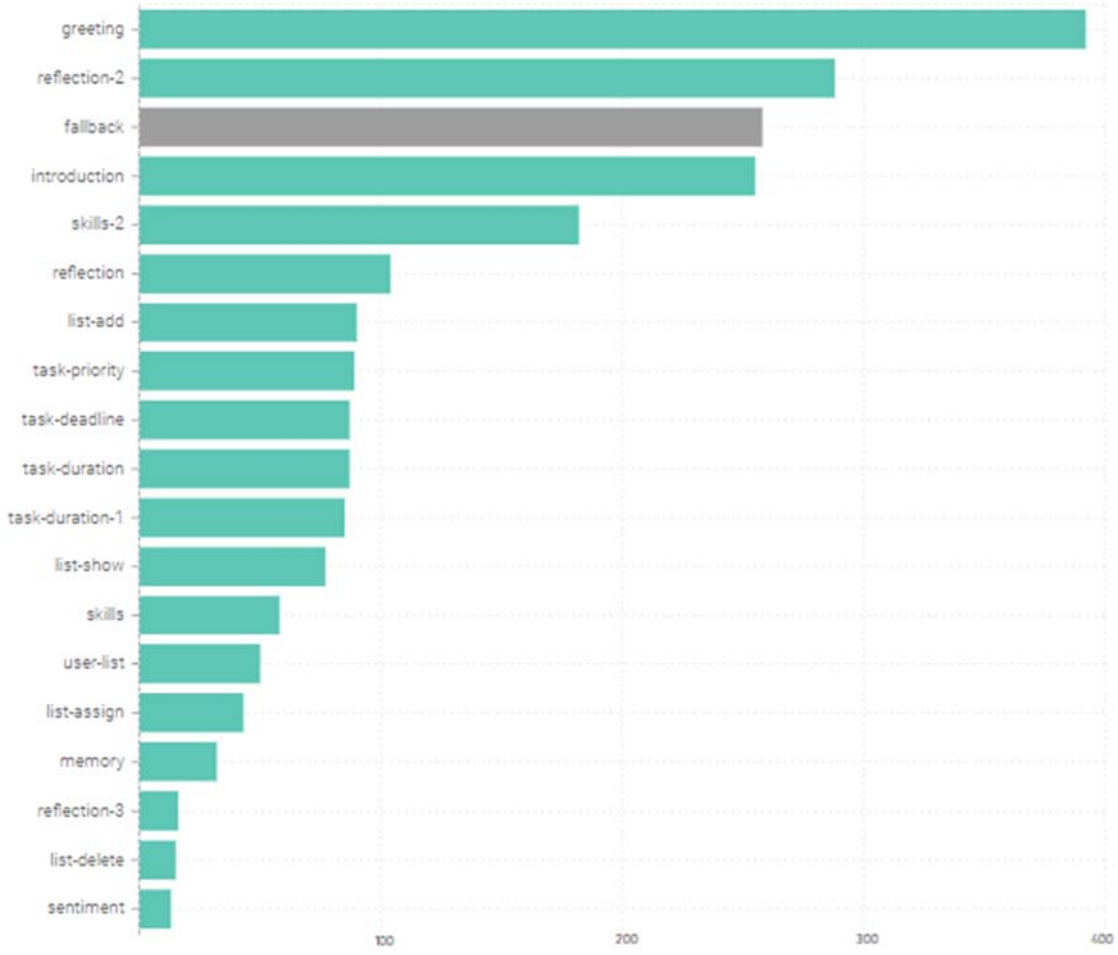


Figure 7-7: Skills Usage

Reflector implemented several skills as explained in chapter 5.5. In the previous illustration, Figure 7-7, the list of skills, and its activation numbers are shown in descending order. Inherently, on top of the statistics, the greeting-skill is listed with a total activation count of 392. The second most activated skill is the reflection-skill, which is activated after the user actively starts a reflection run by typing “Let me reflect on my activities”. The total activation count regards to 288 times. The fallback-skill is located on the third-most position and has been called 258 times. Every time the chatbot cannot understand an input message of a user or does not have implemented a skill for the user’s intent, the fallback skill is activated. About 250 times, the introduction skill has been executed. “Skills-2” is referring to the explanatory skill that lists all possible capabilities of Reflector. The skills “list-add”, “task-priority”, “task-deadline”, “task-duration” and, “task-duration-1” are following. Logically they belong to the same skill execution for creating a

task. They were executed approximately 90 times. The reason for splitting the skill execution into five separate skills was the simplification by using the framework-specific programming pattern of requirements. The skill “list-show” has been triggered 77 times. The skill “show all users” of Reflector has been activated 50 times.

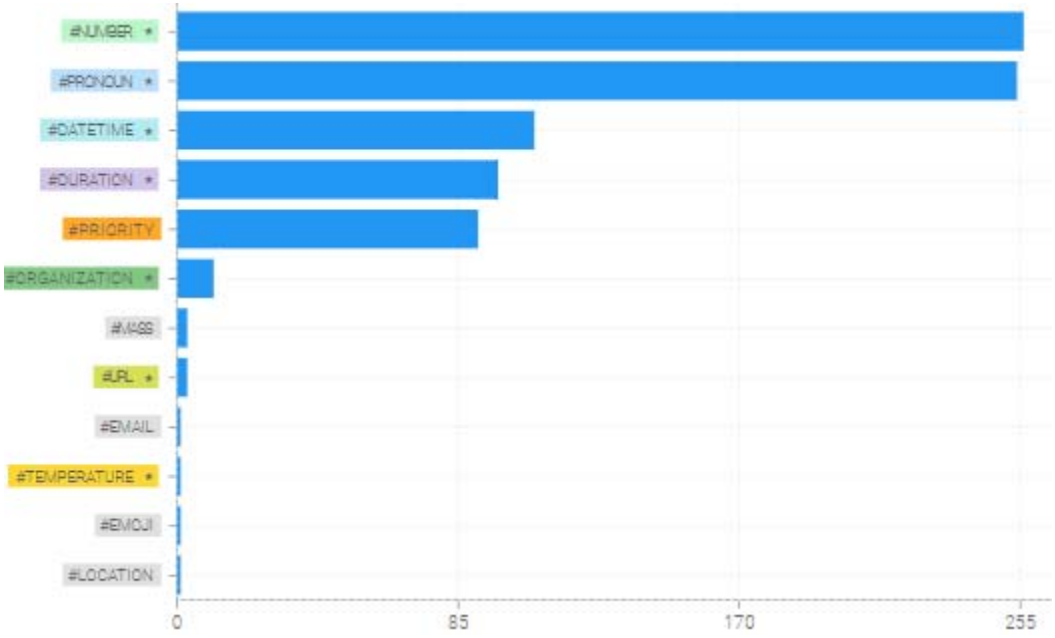


Figure 7-8: Entities Usage

The figure above lists the entities that have been detected by the framework in sentences that have been sent over telegram. On top of the most identified labels, the number and the pronoun entities have been detected. Both labels are encountered during 12th of August, and the 1st of September over 250, where the number-entity exact count was 256 and the pronoun entity exact count was 251

The Date Time entity has been discovered 108 times. The duration was successfully recognized 97 times.

Furthermore, the priority entity was recognized 91 times. At the bottom with irrelevant count numbers, the organization entity was recognized 11 times. All other entities were not used more than five times or more.

2.) How did the participants perceive Reflector and how helpful is Reflector for Task & Time management?

Based on the final interview, one participant said that the chatbot encouraged him/her to “think about task planning in the morning”, because of the daily reminders of Reflector send via Push-Notifications.

Participants also stated that they would have started to chat with reflector during short breaks of their working hours. Half of the participants said that the questions were conclusive and helped them thinking about interruptions. However, participants also said that a task list managing functionality is not very useful on a conversational chatbot as they did not maintain any tasks on the chatbot.

One participant said that the notifications were a good reminder to start reflecting. However, the notifications should be adaptable and not fixed, as well as turned off. One participant said that it [the chatbot] is a good thing because it can be used during commuting with the tramway after work, which supports the feeling of control over their daily achievements. One participant said that the chatbot helped him to train to assess the time for a task more precisely.

The samples of question and answer pairs are gathered from conversations with the chatbot. The answers are from participants from the field study. The questions originate from the predefined pool of activity-related questions. In the underlying text snippets, one can see a series of dialogues.

Table 4 represents a coherent conversation that has been made with Reflector. Within the table, a dialogue about the activities and the reflection upon is visualized.

Table 4: : Coherent conversation of task achievement (23rd of August)

Conversation Protocol
Reflector: Please answer each question as detailed as possible.
Reflector: Ok? Let's start!
Reflector: What tasks are you planning for today?
User: I make a web service connection between a customer system and a internal system.
Reflector: How can you make your daily ToDo's more pleasant?
User: Maybe, look up the necessary authentication properties before testing the connection
Reflector: What were some of the most satisfying moments for you at work yesterday?
User: When I have finished to migrate data collections into a production system
Reflector: What did you learn from these activities?
User: Mass data collections are represented in relational databases or NoSql databases,
Reflector: Could you describe a strategy for approaching unpleasant tasks?
User: To split a task into smaller pieces
Reflector: Will you do anything differently, if you do this or a similar task, in the future?
User: Excel spreadsheets are useful for saving data temporarily, but not for a continuous
Reflector: Great!! You have finished your reflection

Table 5: Coherent conversation of task achievement (22nd of August)

Conversational Protocol
User: I have done a task
Reflector: Which task do I have to remove? Please tell me the name.
Reflector: Here is your list(of tasks): (List of all tasks)
User: Writing an email
Reflector: I deleted "Writing an email" from your list
Reflector: Well done. You have finished the next task for today!
Reflector: How long did it take You to complete the task?
User: 15 minutes
Reflector: Good Job. You have done your task in the estimated duration.

The following dialogue represents an inconclusive conversation because of the incomplete response path. In this case, the conversation flow is linear and responses, where no interruptions happened during the office working hours, are ignored. The question was: "When and where was the last time something interrupted you?". Since the participant response is "I did not face an interruption", the expected behaviour would be not questioning further for interruptions. However, the negative response path has not been developed, and as a result, the conversation lacks flexibility.

Table 6: Inconclusive conversation regarding fragmentation(18th of August)

Conversational Protocol
Reflector: When and where was the last time something interrupted you?
User: I did not face an interruption
Reflector: Was the interruption from a source within the context of your original work?
User: /
Reflector: How urgent was the interruption?
User: Urgent, my Boss asked my something
Reflector: Did you complete the task that was interrupted till the end of the day?
User: No
Reflector: Have you planned times to allow interruptions purposely?
User: I do not understand this question
Reflector: What tasks are you planning for today?
User: Again, Sales enhancement

Another example of an inconclusive conversation is represented in Figure 5-13. The user is asked several questions regarding interruptions. The questions were all answered, but the responses are phrase-like and inexhaustible. For example, the responder wrote, “I do not know” on the question “How urgent was the interruption” aiming at the elaboration of the situation of the participant, which has been asked when and where the last interruption appeared during his work time. Additionally, the participant answered two questions with the word “No”, which does not provide detailed information. The fourth question was answered with, “Can’t say right now”. The question was, “Did you complete the task that was

interrupted until the end of the day?". Consequently, the question targets to an ongoing activity, which has not been terminated yet.

Table 7: Inconclusive conversation regarding fragmentation(22th of August)

Conversational Protocol
Reflector: When and where was the last time something interrupted you?
User: Just now at the office, a support request
Reflector: Was the interruption from a source within the context of your original work?
User: No
Reflector: How urgent was the interruption?
User: I don't know
Reflector: Did you complete the task that was interrupted till the end of the day?
User: Can't say right now
Reflector: Have you planned times to allow interruptions purposely?
User: No

8. Interpretation

Ultimately, the research questions, as defined in the previous chapter, are discussed in this chapter. Unsurprisingly, this research includes limitations, which are described in section 8.4.

8.1 Research Question 1a – Tools & Behaviour

1a) *What tools and which behaviours of Task & Time Management can be examined in the target user group*?

In general, the Task & Time management behaviour of the participants is not extraordinary, resulting from the discussion in section 7.1. The Task & Time management of a cloud developer is based on planning appointments, prioritizing customer-directed activities and achieving project termination goals. The challenges are making correct workload estimations and finalizing projects to the termination date or reacting to unexpected incidents or tasks. The universal tool for planning activities daily and weekly is Microsoft Outlook and JIRA, which are standard tools. As we are discoursing software developers, who are creating, deploying and fixing applications, the prioritization of activities is based on business impact and urgency.

The typical Task & Time management process is not surprisingly repetitive and cyclic, starting with the acquisition of tasks from different sources. The prioritization of tasks is carried out with the impact in business processes as well as project deadlines as criteria. Before a task is executed, a feasibility check and the acquisition of information is happening. The realization of the task is officially achieved by the signing of a hand-over protocol. What is exceptional, there is no monitoring behaviour or retrospection on completed implementation projects in a team or as an individual. As soon as a task is completed, the participants are resuming with planning new activities instead of reviewing or monitoring the achieved project implementation. Concerning Pre-Questionnaire results, in which only 2 participants think, that reflection is often happening in projects, whereas 6 participants are facing this question neutral as well as 1 participant disagrees, confirms this

observation partly. Reflective behaviour of revisiting experiences in the past is missing and would improve the confidence of participants.

8.2 Research Question 1b – Application & Strategies

1b) *How fragmented is the daily work of the target user group*?

For many participants, the daily work consists of various projects with numerous different domains and requirements. The reactions of participants, when presenting visualizations of their task fragmentation, was usually indistinguishable. The participants were impressed to see their task fragmentation and the high frequency in which internal and external interruptions affect their working flow. Furthermore, participants reported Manic Time as a helpful tool to track down the time for work and to get an overview of the productive time of concentrated work versus the time that has been inactive or absent, not sitting in front of the desktop and actively using applications. Increasing awareness about time fragmentation could help users to assess their time use better referring to Pammer et al. (2015).

In Table 3, an example of task fragmentation is laid out. The participants median values for average application usage times are in a time span of five seconds. The span amounts to only five seconds for all ten participants, whereas the shortest median duration a participant remains at the same application windows is ten seconds. Even though, the same context of working activity might require multiple applications, for example, coding with an Integrated Deployment Environment (IDE) app and a browser to backcheck the source code. The time a participant keeps focusing on a single application without switching between separate applications is short. The target group median focusing on an application window without changing to another app is below 15 seconds.

These results are novel, as there are no field studies outside research labs with conversational agents for Task & Time management.

As for many reasons, a smart and sustained way of managing over resources can substantially influence life. People can become aware of spending their time more wisely. The first step to behavioural change is to observe and understand the status quo. However, a vast amount of people does not keep on track, adapt,

and change their behaviour regularly. Conversational agents can animate people to engage with individual problems and help them with their work towards personal goals. In the end, a process of changing bad into good behaviour can lead to better overall well-being.

8.3 Research Question 2 – Usage & Acceptance

2) How extensive has "Reflector" been used, and how helpful did the target user group find the chatbot?

The conversational agent has been used by the participants moderately over the field study time span. The statistics show a decent number of messages and conversations. On the one hand reflector's pro-active reminders were recognized negatively, they interrupted users. On the other hand the Push-Notifications were positively related for reflection reminders.

Unluckily, features for task adding and deleting were not accepted well by the participants. An enhanced way of managing tasks can increase the perceived control of time, which consequently affects every employee in a way, that they experience less stress and a higher feeling of well-being.

Based on the interviews, many participants have reacted positively on the chance to reflect on projects, activities and interruption of their working hours. Participants were able to revisit valuable teachings or learnings. By asking employees about their long-term goals, they could also check and prove if their activities are along with their overall long-term goals.

In the rigorous timeline of projects, which is affecting the employees Task & Time management skills, the opportunity to reflect on the achievements and the activities in the past is crucial. Reflector's questions did help some of the target group to revisit experiences better and this might have helped them to gain confidence in their profession.

8.4 Limitations

The design, implementation & testing was utterly conducted by the author and did consume a significant amount of resources for this master thesis. The risk of interrupting users with Reflector's Push-Notifications was not considered well enough and would a more balanced approach would be necessary in further implementations, referring to Liao et al.(2016).

People usually are communicating by speaking, listening, making gestures, sign languages as well as different kinds of text. The chatbot did support text-based input, but the support of a speech-based chat modality limited the way of users to communicate with Reflector.

The study participants did use their smartphone and their desktop pc as communication devices. Manic time recordings could not track down conversations from smartphones or tablets. The monitoring features of SAP Conversational AI could provide only aggregations of usage data.

In Manic Time, users could label their working applications with different projects. That would have been helpful for the evaluation since participants could effectively connect their recorded time slots to a customer project and observe the time that is remaining in the project.

Another limitation of this thesis may be the quality of data extracted from the chatbots conversations. The conversations were held by the participants, who contributed to the field study in their working time. However, phrase-like or incomplete messages, poor discipline in talking with Reflector must be taken into consideration.

9. Conclusion

Concluding, the goal of this thesis was to investigate and develop a conversational agent for Task & Time management. The two research questions were:

> 1a) *What tools and which behaviours of Task & Time Management can be examined in the target user group*?

The Task & Time management behaviours that have been observed during the field study were unsurprising, considering the study context and the study environment. A lack of setting of long-term goals and a lock of reflection on positive or negative aspects in completed projects has been determined.

> 1b) *How fragmented is the daily work of the target user group*?

The fragmentation of application usage during working hours is significant, as the duration of the participants, empirically measured, focusing on a single application diverges between 10 and 15 seconds in median values. This is novel, as there are no field studies outside research labs with conversational agents for Task fragmentation in this context.

> *2) How extensive has Reflector been used, and how helpful did the target user group find the chatbot*?

Reflector has been used moderately during the field study. However, the participants reported that the chatbot helped to reflect more often.

Overall, study results were affirmative. Given the above limitations, it would be interesting for product development to use the knowledge of constructing a conversational agent for the development of a chatbot in the customer service area.

References

Adamson, D., Dyke, G., Jang, H., & Rosé, C. (2014). Towards an Agile Approach to Adapting Dynamic Collaboration Support to Student Needs. *International Journal of Artificial Intelligence in Education*, S. 92-124. doi:10.1007/s40593-013-0012-6

Amabile, T. M., & Kramer, S. J. (2011). The power of small wins. In *Harvard Business Review* (S. 70-80).

Bannon, L. A. (1983). Evaluation and analysis of users' activity organization. S. 54-57. doi:10.1145/800045.801580

Bellotti, V., Dalal, B., Good, N., Flynn, P., Bobrow, D. G., & Ducheneaut, N. (2004). What a To-Do: Studies of Task Management Towards the Design of a Personal Task List Manager. *Proceedings of the 2004 Conference on Human Factors in Computing Systems.* Vienna. doi:10.1145/985692.985785

Britton, B., & Tesser, A. (September 1991). Effects of Time-Management Practices on College Grades. *Journal of Educational Psychology*. doi:10.1037/0022-0663.83.3.405

Claessens, B. J., van Eerde, W., Rutte, C. G., & A. Roe, R. (February 2007). A review of the time management. *Personnel Review, Vol. 36 Issue: 2,*.

Czerwinski, M., Horvitz, E., & Wilhite, a. S. (2004). A diary study of task switching and interruptions. S. 175-182. doi:10.1145/985692.985715

Di Stefano, G., Gino, F., Pisano, G. P., & Staats, B. R. (2014). Learning by Thinking: How Reflection Aids Performance. *The power of reflection*. doi:10.2139/ssrn.2414478

Dohrn-van Rossum, G. (1996). *History of the Hour: Clocks and Modern Temporal Orders.* Chicago: Univ. of Chicago Press.

Drucker, P. F. (1967). *The Efiective Executive.*

Durov, P., & Durov, N. (2019). *Telegram*. Von https://telegram.org abgerufen

Edinburgh, U. o. (2019). *The University of Edinburgh*. Von https://www.ed.ac.uk/reflection/reflectors-toolkit/reflecting-on-experience/gibbs-reflective-cycle abgerufen

Eerde, W. V. (2003). Procrastination at Work and Time Management Training. (PubMed, Hrsg.) *The Journal of Psychology Interdisciplinary and Applied*. doi:10.1080/00223980309600625

Eilam, B., & Aharon, I. (July 2003). Students' planning in the process of self-regulated learning. *Contemporary Educational Psychology*, S. 304-334. doi:10.1016/S0361-476X(02)00042-5

Fessl, A., Wesiak, G., Rivera-Pelayo, V., Feyertag, S., & Pammer-Schindler, V. (2017). In-App Reflection Guidance: Lessons Learned. *IEEE Transactions on Learning Technologies*. doi:10.1109/TLT.2017.2708097

G, G. (1988). *Learning by Doing: A guide to teaching and learning methods. .* Oxford: Further Education Unit. Oxford Polytechnic.

Gaesser, A., & McNamara, D. &. (2005). Scaffolding Deep Comprehension Strategies ThroughPoint&Query, AutoTutor, and iSTART. *Educational Psychologist, Vol. 40, No. 4*, S. 225-234.

Graesser, A. C. (2001). Intelligent tutoring systems with conversational dialogue. *AI Magazine*, S. 39–51. doi:https://doi.org/10.1609/aimag.v22i4.1591

Graesser, A., & McNamara, D. &. (2005). Scaffolding Deep Comprehension Strategies ThroughPoint&Query, AutoTutor, and iSTART. *Educational Psychologist*, S. 225-234. doi:https://doi.org/10.1207/s15326985ep4004_4

Harper, R. &. (2008). *Being Human: Human-Computer Interaction in the Year 2020.*

Holtzblatt, K., & Beyer, H. (2016). *Contextual Design.* Morgan Kaufmann.

Horvitz, E., & Iqbal, S. T. (2007). Disruption and recovery of computing tasks: field study, analysis, and directions. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. doi:10.1145/1240624.1240730

Hudson, J., Christensen, J., Kellogg, W., & Erickson, T. (2004). I'd Be Overwhelmed, But It's Just One More Thing to Do:. *Availability and Interruption in Research Management*.

Kocielnik, R., Avrahami, D., Marlow, J., Lu, D., & Hsieh, G. (2018). Designing for Workplace Reflection: A Chat and Voice-Based Conversational Agent. doi:10.1145/3196709.3196784

Lakein, A. (1973). *How to Get Control of Your Time and Your Life.*

Leshed, G., & Sengers, P. (2011). "I lie to myself that I have freedom in my own schedule": Productivity tools and experiences of busyness. *Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2011.* Vancouver, BC, Canada.

Liao, Q. V., Davis, M., Geyer, W., Muller, M., & Shami, S. (2016). What Can You Do?: Studying Social-Agent Orientation and Agent Proactive Interactions with an Agent fo Employees. In P. o. systems (Hrsg.)., (S. 264–275).

Macan, T. (June 1994). Time Management: Test of a Process Model. *Journal of Applied Psychology*, S. 381-391. doi:10.1037/0021-9010.79.3.381

Mark, G., Gonzalez, V. M., & Harris, J. (2005). No Task Left Behind? Examining the Nature of Fragmented Work. *Take a Number, Stand in Line (Interruptions & Attention 1)*. doi:10.1145/1054972.1055017

Mark, G., Gudith, D., & Klocke, U. (April 2008). The Cost of Interrupted Work: More Speed and Stress. *Don't Interrupt Me*. doi:10.1145/1357054.1357072

McCay, J. T. (1968). *The Management of Time.*

Microsoft Corporation. (2019). Von Microsoft Azure: https://azure.microsoft.com/en-us/overview/what-is-azure/ abgerufen

Nardi, B. A. (1996). *Context and Consciousness - Activity Theory and Human-Computer Interaction.* The MIT Press.

Nikabot. (2018). (Impossible Comp) Von https://www.nikabot.com abgerufen

Pammer, V., & Bratic, M. (2013). Surprise, Surprise: Activity Log Based Time Analytics for Time Management. *Computer-mediated reflection in the Workplace.* doi:10.1145/2468356.2468395

Pammer, V., Bratic, M., Feyertag, S., & F. ,. (2015). The Value of Self-tracking and the Added Value of Coaching in the Case of Improving Time Management. *Design for Teaching and Learning in a Networked World.* Springer, Cham. doi:10.1007/978-3-319-24258-3_41

Ruan, S., Jiang, L., Xu, J., Joe-Kun Tham, B., Qiu, Z., Zhu, Y., . . . Brunskill, E. &. (2019). QuizBot: A Dialogue-based Adaptive Learning System for Factual Knowledge. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*.

SAP Help Portal. (November 2019). Von SAP Conversational AI: https://help.sap.com abgerufen

Skinner, D. (2005). Does Time Management Training Work? An Evaluation. *International Journal of Training and Development*. doi:10.1111/j.1468-2419.2005.00226.x

Status Hero. (2018). (8012 Labs) Von https://statushero.com abgerufen

Teevan, J., Liebling, D., & Lasecki, W. (2014). Selfsourcing personal tasks.

List of illustrations

List of tables

List of abbreviations

TTM……...Task and Time Management

CD……...Contextual Design

SAPCAI…… SAP Conversational AI

NLP……Natural Language Processing

HTTP……Hyper Text Transfer Protocol

API……..……Application Programming Interface

IDE……..……Integrated Deployment Environment

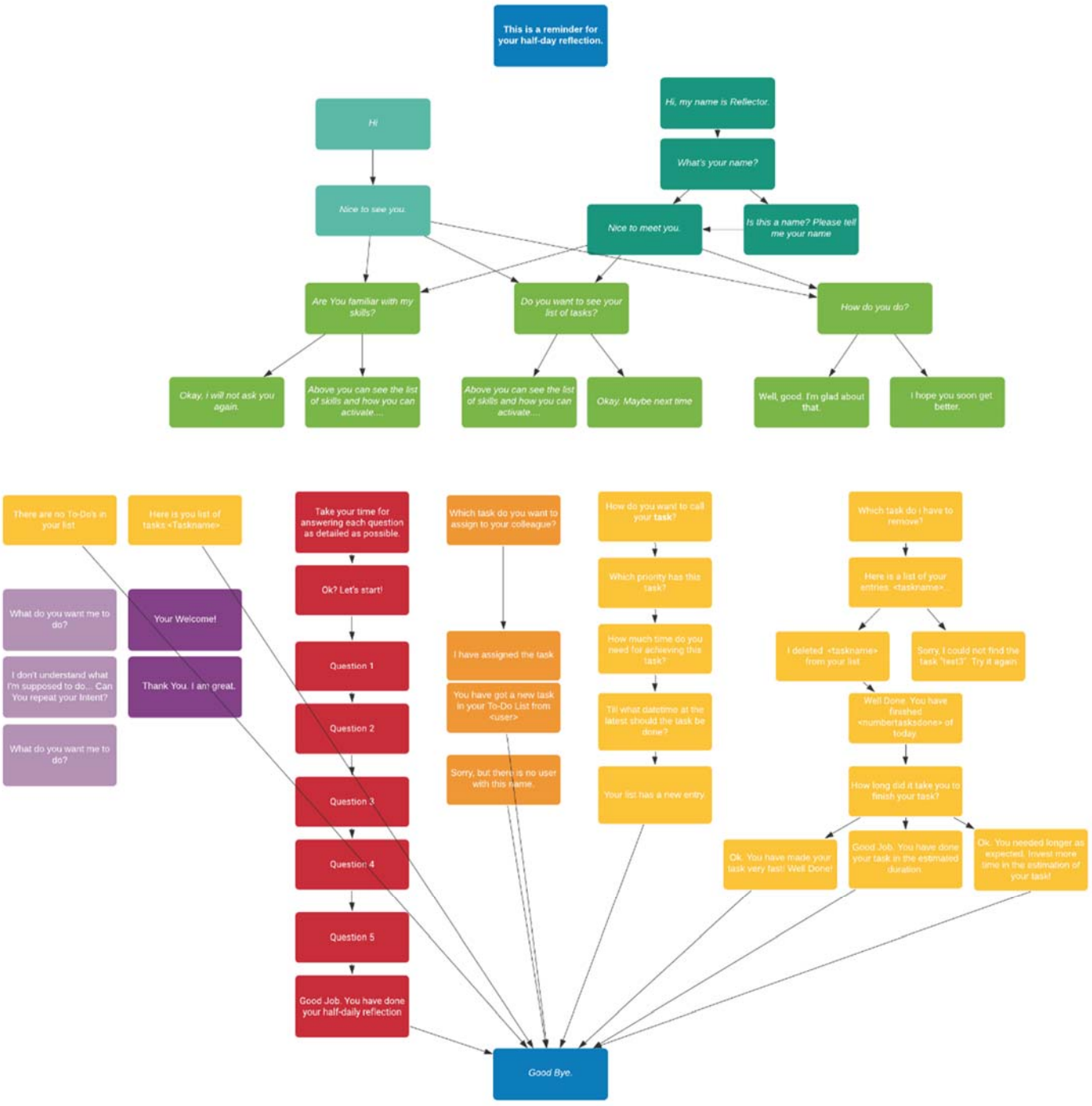
PCode……Participant Code

IoT……Internet of Things

REST……..Representational State Transfer

Appendix


Research
Design
Evaluation
Writing & Presentation
Domain
User
Application
Literature summary
Study User Groups
Available Technology
Release co-written ebook with partnership
Persuasion of Users
Weekly Scrum Sprints
Build up Hypothesis
Evaluation of User Time Management
Fix outstanding bugs
Promote launch via social media
Diagram Legend
Domain
User
Application


This is a reminder for your half-day reflection.
Hi
Nice to see you.
Hi, my name is Reflector.
What's your name?
Nice to meet you.
Is this a name? Please tell me your name
Are You familiar with my skills?
Do you want to see your list of tasks?
How do you do?
Okay, I will not ask you again.
Above you can see the list of skills and how you can activate....
Above you can see the list of skills and how you can activate....
Okay. Maybe next time
Well, good. I'm glad about that.
I hope you soon get better.
There are no To-Do's in your list
Here is you list of tasks:<Taskname>
What do you want me to do?
I don't understand what I'm supposed to do... Can You repeat your Intent?
What do you want me to do?
Your Welcome!
Thank You. I am great.
Take your time for answering each question as detailed as possible.
Ok? Let's start!
Question 1
Question 2
Question 3
Question 4
Question 5
Good Job. You have done your half-daily reflection
Which task do you want to assign to your colleague?
I have assigned the task
You have got a new task in your To-Do List from <user>
Sorry, but there is no user with this name.
How do you want to call your task?
Which priority has this task?
How much time do you need for achieving this task?
Till what datetime at the latest should the task be done?
Your list has a new entry.
Which task do i have to remove?
Here is a list of your entries: <taskname>...
I deleted <taskname> from your list
Sorry, I could not find the task "test3". Try it again
Well Done. You have finished <numbertasksdone> of today.
How long did it take you to finish your task?
Ok. You have made your task very fast! Well Done!
Good Job. You have done your task in the estimated duration
Ok. You needed longer as expected. Invest more time in the estimation of your task!
Good Bye.

```
[
  {
    "answer": "----Seperation-----",
    "question": "Day 2019-08-05: "
  },
  {
    "answer": "I have created a plan on paper on monday morning as I do every week for preparation purposes.",
    "question": "When and how did you create your weekly plan the last time?"
  },
  {
    "answer": "I am working on 3-4 customer projects, where every project represents a bigger task for me to achieve.",
    "question": "How many different tasks(approx.) do you write on your weekly/daily Todo list?"
  },
  {
    "answer": "No, i have not scheduled every task.",
    "question": "Do you have scheduled all tasks in your calendar?"
  },
  {
    "answer": "Well, i could arrange time slots for my customer projects in order to avoid unpleasent interruptions.",
    "question": "How can you\nmake your weekly ToDo's more pleasant for you  to do?"
  },
  {
    "answer": "Collecting the necessary numbers in preparation steps",
    "question": "You have noted the task: Writing Expense Report .Could you take away anything valuable for the future from doing this task?"
  },
  {
    "answer": "I underestimated the time for querying the numbers",
    "question": "Was there anything that made you feel \nhappy/unhappy when doing \n<task>? What was it? How can you\nlearn from it?"
  },
  {
    "answer": "----Seperation-----",
    "question": "Day 2019-08-06: "
  },
  {
    "answer": "I want to write  a short summary of my activities yesterday and i want to make progress with my customer project ",
    "question": "What tasks are you planning for today?"
  },
  {
    "answer": "To make a brainstorming about all relevant information and gather it on a piece of paper",
    "question": "How can you\nmake your daily ToDo's more pleasant to do?"
  },
  {
    "answer": "When i have finished to make the screenshot gallery of one of my projects",
    "question": "What were some of the most\nsatisfying moments at work for you yesterday?"
```

```
    },
    {
        "answer": "Motivation does help a lot when working on a single task",
        "question": "What have you learned from these activities?"
    },
    {
        "answer": "To cut it into smaller chunks of tasks and therefore reduce the com-
plexity of it",
        "question": "Could you describe a strategy for approaching unpleasent tasks?"
    },
    {
        "answer": "Maybe i try to work with multiple displays to have a better overview of
all screenshots and saved photo files.",
        "question": "Will you do anything differently if you do this task, or a similar task, in
the future?"
    },
    {
        "answer": "----Seperation-----",
        "question": "Day 2019-08-07: "
    },
    {
        "answer": "Canyon integration",
        "question": "What tasks are you planning for today?"
    },
    {
        "answer": "Have a clear calendar distinction",
        "question": "How can you\nmake your daily ToDo's more pleasant to do?"
    },
    {
        "answer": "When i have finished my work",
        "question": "What were some of the most\nsatisfying moments at work for you
yesterday?"
    },
    {
        "answer": "That the early bird catches the worm",
        "question": "What have you learned from the activities yesterday?"
    },
    {
        "answer": "Think it trough before starting it",
        "question": "Could you describe a strategy for approaching unpleasent tasks?"
    },
    {
        "answer": "Make a list of expectations",
        "question": "Will you do anything differently if you do this task, or a similar task, in
the future?"
    },
    {
        "answer": "----Seperation-----",
        "question": "Day 2019-08-08: "
    }]
```


Example Reflection Log of week 1 extracted from https://caiserver.azurewebsites.net: