Bachelor's thesis at Institute of Computer Science at Freie Universität Berlin

Human-Centered Computing (HCC)

Design, Implementation, and Evaluation of a Cross-Platform Mobile Application for Experiments in Research on Electronic Brainstorming

Zurab Chkhitunidze

Supervisor: Alexa Schlegel

Primary Reviewer: Prof. Dr. Claudia Müller-Birn

Secondary Reviewer: Prof. Dr. Lutz Prechelt

Berlin, 26.05.2021

b

Eidesstattliche Erklärung

Ich versichere hiermit an Eides Statt, dass diese Arbeit von niemand anderem als meiner Person verfasst worden ist. Alle verwendeten Hilfsmittel wie Berichte, Bücher, Internetseiten oder ähnliches sind im Literaturverzeichnis angegeben, Zitate aus fremden Arbeiten sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

Berlin, 26. Mai 2021

Zurab Chkhitunidze

Abstract

Electronic brainstorming has been a topic of interest for researchers in the past few decades. Studies show that in some cases using computers for idea generation and problem-solving delivers promising results. Considering the capabilities of modern smartphones and the fact that they are almost instantly accessible to their owners most of the time, interesting subjects of research might be the effectiveness of different strategies of "mobile brainstorming" and the effects of involving smartphone features like notifications into the process. A mobile app designed for conducting experiments in this field might be helpful for researchers.

The goal of this work was to create a prototype of such an application following human-centered design principles focusing on participants of experiments. This work describes the development process of this prototype as a cross-platform mobile application through iterations of the design, implementation, and evaluation cycle. The app provides various options to researchers for conducting experiments with different rules and settings to explore the effectiveness of different variations of electronic brainstorming on mobile devices.

The results of this work include a prototype in the form of a cross-platform mobile app, data gathered from users through interviews and tests, an overview of the problems that occurred during this work, and an outlook for future work.

Zusammenfassung

Elektronisches Brainstorming war in den letzten Jahrzehnten ein interessantes Thema für Forscher. Studien zeigen, dass der Einsatz von Computern zur Ideenfindung und Problemlösung in einigen Fällen vielversprechende Ergebnisse liefert. In Anbetracht der Fähigkeiten moderner Smartphones und der Tatsache, dass sie für ihre Besitzer die meiste Zeit fast sofort erreichbar sind, könnten die Effektivität verschiedener Strategien des "mobile Brainstorming" und die Auswirkungen der Nutzung von Smartphone-Funktionen wie Benachrichtigungen in den Prozess interessante Forschungsthemen sein. Eine mobile App für die Durchführung von Experimenten in diesem Bereich könnte für Forscher hilfreich sein.

Das Ziel dieser Arbeit war es, einen Prototyp einer solchen Anwendung zu erstellen, der den Prinzipien des human-centered Designs folgt und sich auf die Teilnehmer der Experimente konzentriert. Diese Arbeit beschreibt den Entwicklungsprozess dieses Prototyps als plattformübergreifende mobile Anwendung durch Iterationen des Design-, Implementierungs- und Evaluierungszyklus. Die App bietet Forschern verschiedene Optionen für die Durchführung von Experimenten mit unterschiedlichen Regeln und Einstellungen, um die Effektivität verschiedener Varianten des elektronischen Brainstormings auf mobilen Geräten zu untersuchen.

Die Ergebnisse dieser Arbeit sind ein Prototyp in Form einer plattformübergreifenden mobilen App, Daten, die von Benutzern durch Interviews und Tests gesammelt wurden, ein Überblick über die Probleme, die während dieser Arbeit aufgetreten sind und ein Ausblick auf zukünftige Arbeiten.

Contents

List of Figures

List of Tables

1 Introduction

Brainstorming [APAa] is a widely used idea generation and problem-solving strategy usually performed in a group setting. Since popularization of this strategy by Alex F. Osborn [Osb53], several modifications [Bra] of this strategy were developed. The effects of using computers in the idea generation process have been researched and discussed in several studies [MIT, PBWK19, Lub05]. As mobile technology evolves and becomes more accessible for everyone, electronic brainstorming [apab] using smartphones could be a topic of interest for more scientific research. Considering features of modern smartphones and the fact that they are almost instantly accessible for users, "mobile brainstorming" may have a different impact on the brainstorming process and the results in comparison to electronic brainstorming using stationary computers. For further research on this topic, a mobile application might be useful, with the help of which researchers would be able to conduct experiments with different settings (e.g. duration of the brainstorming session can be set to several days or reminders to add new ideas can appear every day at certain time).

This work describes the process of development of a prototype of such app, following human-centered design [HCD] principles. This application is aimed to provide researchers with flexibility to conduct different experiments with different groups of participants. Researchers can create and manage experiment projects with custom settings. They can invite participants to the experiments by distributing invitation codes to them. The app collects essential data during experiments and provides it to researchers, who can export it for analysis in other tools.

The process of ideation during experiments using the app takes place asynchronously. Researchers may not assist the participants during the whole time of an experiment, and the facilitation [CDD16] of the brainstorming process can be partially or fully managed by the app based on settings chosen by researchers during the creation of the app.

The main focus of this work lies on the participants of experiments. Several interviews and usability tests helped to identify user needs and potential problems. First software requirements emerged from exploratory interviews and analysis of similar software helped to design a low-fidelity prototype. During the tests of this prototype, users provided feedback for further development and improvements. The next phase of this work included developing the first version of the app as a high-fidelity prototype as well as further usability tests. Finally, in a field deployment [SHNT14], participants used the software in order to generate ideas to solve a real-life problem.

1.1 Terms

Some of the terms used in this work are defined below.

electronic brainstorming is an activity in which software is used for idea generation and problem-solving [apab].

brainstorming facilitrator is a role in the brainstorming strategy, who leads and coordinates the brainstorming sessions. In the case of electronic brainstorming, the software might perform some or all of the facilitator's tasks.

ideation is a process of forming ideas [ide].

Ideation-App is used as a name of the cross-platform mobile application that is being developed throughout this work.

experimenter (also *admin*) is a user role in the Ideation-App. Experimenters are researchers who create experiment projects with different settings and rules within the app and have access to the results of experiments.

experimentee is another user role in the Ideation-App. Experimentees are people who participate in experiment projects managed by experimenters using the software on their mobile devices.

1.2 Goals

This work aims to develop a prototype of a mobile application designed for deployment in experimental studies on electronic brainstorming with the help of mobile devices. The goal is to implement a cross-platform application that runs on Android and iOS platforms using methods described in the Section 1.3.

Even though the application consists of two different parts (interface for researchers who conduct such brainstorming experiments and an interface for experiment participants), this work's primary focus lies in designing the user interfaces and user experience for participants.

In this context, the goal is to develop a prototype focusing on usability in each step of the design cycle: designing, implementing, and evaluating.

1.3 Methodology

This work follows the principles of user-centered approach [JP15, p. 327-330]. This approach includes an early focus on users and tasks, empirical measure-

ments, and iterative design. The thread of work consisted of cycles of designing, testing, measuring, and redesigning.

The beginning of the work implied gathering the first requirements by collecting and analyzing data from interviews and researching similar software. The design of the low-fidelity prototype resulted from these requirements. Data gathered during the evaluation of the prototype was used for designing the next prototype.

The high-fidelity prototype of the Ideation-App is a cross-platform mobile app. Its design derived from the analyzed results of the evaluation. The evaluation of the high-fidelity prototype took place in the form of a field study. The analysis of the evaluation results was the basis for suggestions for further development.

1.4 Related Work

Leonard Przybilla et. al. provide promising results in their work **"Machines as Teammates in Creative Teams: Digital Facilitation of the Dual Pathways to Creativity"** [PBWK19] in using digital facilitation via algorithms and automated systems for creative idea generation in teams. In their experiment, where they conducted tests with human and digital facilitators, they found out that the digital facilitator did not perform significantly differently compared to the human one considering objective criteria and satisfaction. They mention that the type of instructions provided by a facilitator affects output quantity, which moreover correlates with quality. In addition, rules such as giving time for individual brainstorming without group interaction can alleviate issues of production blocking. Based on these findings, they designed a digital facilitator that is using the following interventions:

- encouraging teams to go for quantity
- providing teams with general rules for brainstorming
- giving teams time for individual brainstorming in order to reduce effects of production blocking

The high-fidelity prototype of the Ideation-App has a facilitation module (described in the Section 3.3.7) that allows some facilitation features during ideation sessions. This module provides options to experimenters to prepare studies using any of three interventions mentioned above.

In their work **"Designing Idea Management Tools: Three Challenges"** [IDD18] N. Inie et. al. reveal challenges that designers experience during working with different idea management tools for capturing, storing, retrieving, and

collaborating on their ideas. They identified three major challenges regarding idea management tools:

1. they are rigid in capture medium
2. their interfaces and representations are inflexible
3. instead of ideation they mainly focus on ideas

For each of these challenges, researchers suggest opportunities for novel features or tools. One finding in this work was that the reviewed tools are not actively helping users to revisit their ideas. Authors suggest, for instance, sending push-notifications to encourage users revisiting of the ideas. Another suggestion is to use the advantages of a large touchscreen and many different objects at once. The implementation of the Ideation-App supports these suggestions.

In the Work **"How can computers be partners in the creative process: classification and commentary on the special issue"** [Lub05], Todd Lubart examines different ways in that computers can be involved in creative work: computer as a nanny, pen-pal, coach, colleague. He describes these four possible futures for computers in the field of creativity and discusses issues concerning the creative process.

Design of configurable project features of the Ideation-App allows researchers to implement the approach mentioned in this paper named "computer as nanny". The Authors of the paper mention that the most frequently cited attribute needed for creativity is perseverance, and computers can promote it. They can help users by setting deadlines and thus handling time pressure and keep a project on schedule. Also, computers can remind the user of these deadlines. Another idea is that computers can plan out user's activities to keep blocks of time for creative projects. Another approach called "computer as coach" involves giving users information about the existing techniques to stimulate their creativity. Experimenters can achieve this by creating a new experiment in the Ideation-App by defining the content of notifications or informational pages and popups.

After reviewing literature from the domains of engineering design and psychology, T.J. Howard et. al. propose a creative design process as an integration between the engineering design process and the creative process established from cognitive psychology in their work **"Describing the creative design process by the integration of engineering design and cognitive psychology literature."** [HCD08]. Engineers would be able to properly use creativity tools, processes, and strategies if they understand the linkages in the overall process, according to the authors. Understanding this process can also show where and when resources should be spent to improve creative output and the quality of the designed product. The authors are convinced that clearly defined creative

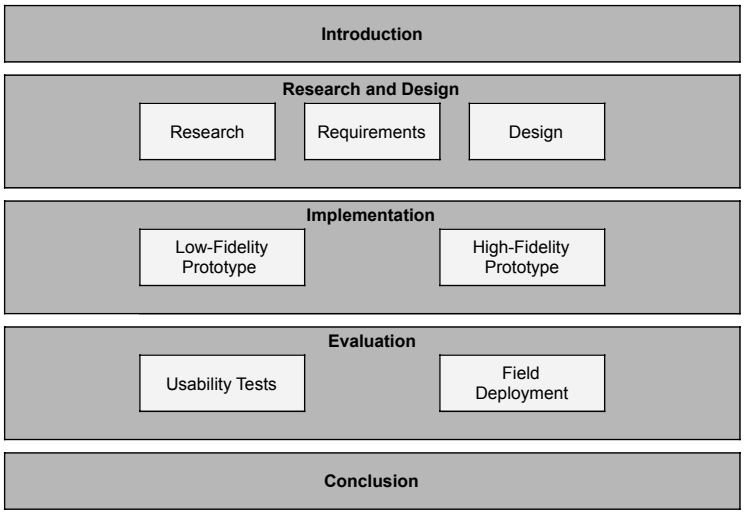
design output consisting of measurable elements (originality and appropriateness) will help researchers to measure the effectiveness of newly proposed creativity tools and methods. The design of user experience of the Ideation-App is partially based on their integrated creative-design process model.

Evaluation of the high-fidelity prototype had a form of a field deployment, using techniques described in **"Field Deployments: Knowing from Using in Context"** [SHNT14]. Field deployments can be used to evaluate encounters with users in extreme situations and everyday life. Researchers who deploy prototypes in the wild can collect data using a variety of techniques. Field deployments could be expensive, need a large number of resources but they mostly lead to collecting valuable data from usage in a natural setting. Section 4.2 describes using this method for evaluating the high-fidelity prototype of the Ideation-App. Section 4.2.3 contains results of this evaluation.

1.5 Structure of the Work

This thesis is structured into chapters thematically, not chronologically. Besides the introduction and conclusion, this work consists of three chapters: Research and design (2), Implementation (3), and Evaluation (4). Chapter 2 describes steps taken during the research and design of the app, including data gathering and defining requirements. Chapter 3 summarizes the process of implementation and describes the components of the software. Chapter 4 presents the methods of evaluation, describes the execution process, and reports the results. Finally, Chapter 5 summarizes the work and results, discussing problems that occurred during this work, and concludes with directions for future work.

Table 1.1: Structure of the Work



2 Research and Design

The purpose of this chapter is to describe methods and actions taken during the research and design phase. It contains the research plan created at the beginning of the research phase of this work and a report on the steps taken following the research plan. These steps include: gathering data through interviews and researching similar software, creating personas, scenarios, use cases. Finally, there are the results of the work made during the research and design phase — a list of requirements and app mockups.

2.1 Research Plan

This research plan sets the roadmap for the research phase of this work and defines fundamental matters of the research. It is a brief description of research goals and methods and the definition of the software's stakeholders.

Stakeholders are people who are affected by the software and directly or indirectly influence its requirements [GK98]. In the case of this work, the software is meant to be used in scientific studies as a tool for conducting experiments with different groups of participants. Therefore stakeholders are scientists (experimenters) responsible for managing experiments and participants (experimentees) who will participate in these experiments. This work focuses mainly on participants.

The goals of the research phase imply understanding stakeholders and gathering applicable information for the software design. The first goal is gathering data on existing practices and experience of users in the field of ideation through interviews. This data helps determine how people develop and share ideas and if they use any particular techniques. The other goal is to create a list of similar software and later analyze them in the research process. A further goal is to gather information about user expectations from a new app and their suggestions for the design.

Methods of data gathering in this work consist of two techniques: semi-structured interviews and analysis of similar and related software. As this work focuses on the experimentees part of the app, in the first step, interviews with people who regularly gather ideas will be held to find out about their experience in the context of idea generation and their suggestions for new software. Then, design solutions of the software that offers services for the idea generation process will be analyzed.

Interviews will provide data on existing practices in the idea generation field and expectations of the new software. For this purpose, the format of interviews will be semi-structured as it is suitable for finding answers to exact

questions (for example, a list of software that interviewees are already using for idea generation) as well as providing exploratory data through open questions that is useful for finding more about user experience and their expectations then through closed questions.

Analysis of similar and related software can start after a list of such software is available. Gathered data from the first interviews should define the first part of the list. More programs for analysis might land on this list after an online research. The goal is to find out about existing solutions for storing, retrieving, sharing, rating, and commenting contributions.

2.2 Data Gathering

This section describes the procedure and results of both methods of data gathering (interviews and researching similar software) used in the early phase of this work. Applying each of these methods and analyzing the results helped to derive the software requirements (presented in the Section 2.6).

2.2.1 Interviews

The goal of the interviews was to find out how people develop ideas in a group and if they use particular techniques. Besides that, the goal was to create a list of software that people use for this purpose. The purpose of creating this list was to analyze the software in the next step. The last goal was to capture their expectations for the future software and their personal reasons for using such an app. To achieve these goals, three interviewees participated in semi-structured interviews [JP15, p. 234-235]. This kind of interview uses both closed and open questions, which enables the collecting of quantitative data such as a list of software tools as well as obtaining qualitative data in the form of, for example, suggestions and recommendations for designing a new tool.

Preparation for the interviews started with creating the consent forms in English and German languages (Appendices 5.1 and 5.2). Then, the questions for the interviews were prepared.

Three potential interviewees (Appendix 5.3: P1, P2, P3) agreed to participate in interviews. In all three cases, interviews were conducted online, through a video call. At the beginning of each interview, participants received information about the purposes of the interview and a brief description of the software.

The next step was to analyze the gathered answers (Appendices 5.4, 5.5, 5.6), to create a list of similar software, and to organize and group qualitative data in an affinity diagram [JP15, p. 234-235]. Affinity diagramming is a common technique used in qualitative analysis. The emerged affinity diagram (Figure 2.1) indicates properties and features for the design of the software.

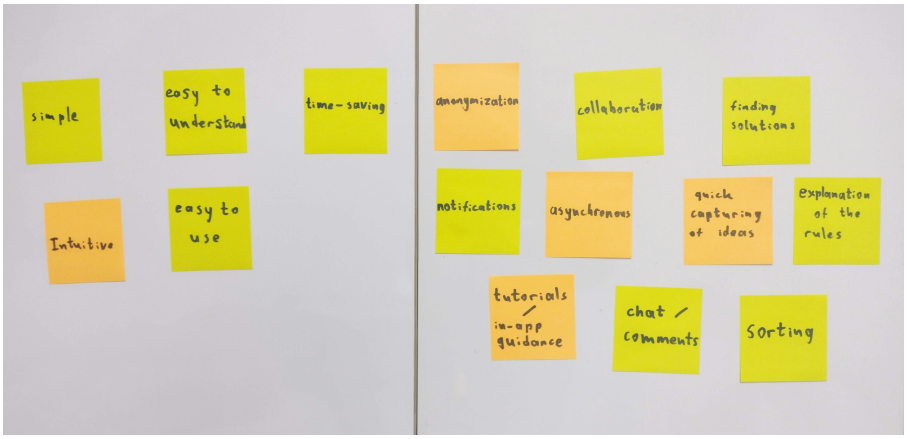


Figure 2.1: Affinity Diagram: Qualitative Data from Semi-Structured Interviews

2.2.2 Researching Similar Software

During interviews, participants mentioned several tools, that they use for processes related to ideation: Google Docs[1], Dropbox[2], Google Keep[3], Google Calendar[4], Skype[5], WhatsApp[6]. An online research revealed that besides the pointed tools, Microsoft To Do[7] is also a widely used tool for storing and sharing ideas, so this software complemented the list.

During the software analysis, the focus was on supporting features that are necessary or helpful for an app, intended for conducted experiments in the field of ideation with groups of participants using their own mobile devices. These features included reminders, collaboration, sorting, online storage, comments, and chat. Table 2.1 is a short, visual representation of the results.

[1]https://www.google.com/docs/about/
[2]https://www.dropbox.com/
[3]https://www.google.com/keep/
[4]https://www.google.com/calendar/about/
[5]https://www.skype.com/
[6]https://www.whatsapp.com/
[7]https://todo.microsoft.com/tasks/

Table 2.1: Researching Similar Software — Features of Different Products

	reminders	collaboration	sorting	online storage	comments / chat
Google Docs	✗	✓	✓	✓	✓
Dropbox	✗	✓	✓	✓	✓
Google Keep	✓	✓	✗	✓	✗
Google Calendar	✓	✓	✗	✓	✗
Skype	✗	✓	✗	✓	✓
WhatsApp	✗	✓	✗	✓	✓
Microsoft To Do	✓	✓	✓	✓	✗

Besides Google Keep, the primary purpose of these apps is not capturing, storing, or sharing ideas. Nevertheless, they offer different sets of features that enable users to utilize them for these purposes, which was the motivation behind choosing these tools for analysis. Neither of these tools, except Google Docs, offer rich or complex features but rather intuitive, basic functionality. All of these tools offer collaboration and online storage features. Some tools lack commenting or chatting features that make it more difficult for users to express their thoughts during collaboration.

Most of these tools have a simple and intuitive design. Material design [Mat] is dominant as a visual design of the user interfaces among the mentioned apps.

Information about the data usage and storage is displayed at least on one of the first screens of each app. All of them ask users for consent to store and use data according to their terms. Users are mostly expected to give their consent by setting marks in one or several checkboxes.

2.3 Personas

Personas [JP15, p. 357-358] describe typical users of the product. They do not represent real people but a synthesis from a number of real users that were involved in data gathering. They are realistic rather than idealized. Personas can be helpful because they can be focused on during a design phase of the software.

Figures 2.2 and 2.3 represent two personas with their pictures, short biographies, characteristics, and their goals and frustrations regarding participating in an app-related study.

The pictures were taken form *this person does not exist* project[8].

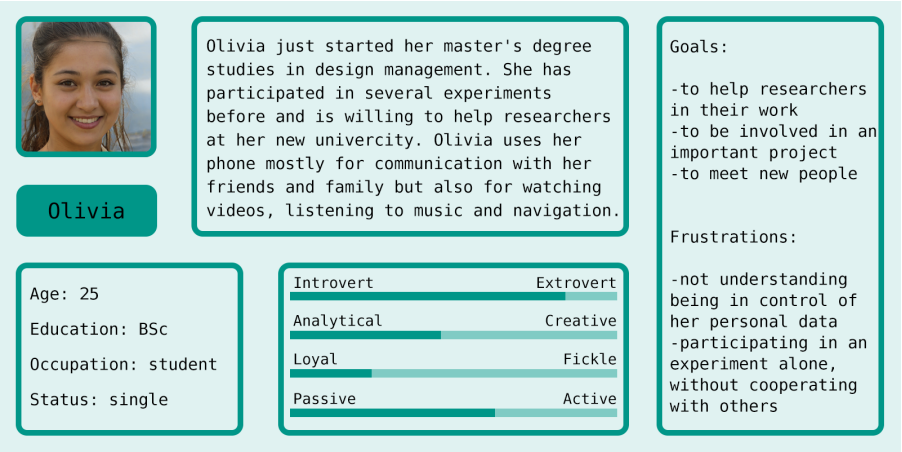


Figure 2.2: Persona — Olivia

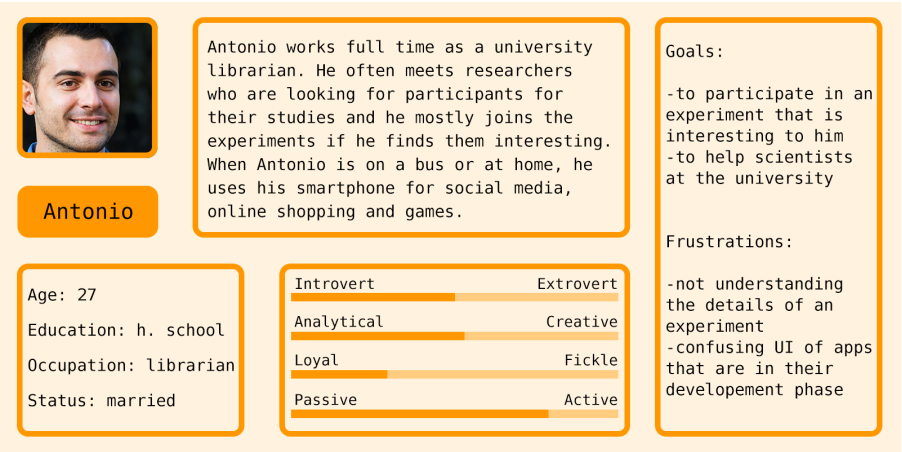


Figure 2.3: Persona — Antonio

2.4 Scenario

A scenario [Car00] is an informal narrative description that intends to describe human activities and tasks in the form of a story. This allows to explore and discuss the contexts, needs, and requirements. Construction of scenarios is often the beginning of setting requirements of a project [JP15, p. 371].

The following scenario introduces a typical situation in which a group of researchers conducts an experiment with participants using the Ideation-App. One of the participants is Olivia, the person behind the persona from the Section 2.3.

[8]https://thispersondoesnotexist.com/

Scenario - a typical experiment

A group of researchers wants to find out how using smartphones for brainstorming impacts the process and results of ideation sessions compared to the traditional approach. For each case, they have a group of experimentees for the first two experiments. Both groups have the same main task: generate the best ideas for a new name for a bike repair shop.

Researchers conduct a brainstorming session with one group using traditional methods. In this case, one of the researchers plays the role of the brainstorming facilitator.

For the second group, they create a new project in the Ideation-App with custom parameters, similar to the one of the first group: the explanation of the process to experimentees should be performed by the app, during the first session, submitted ideas should not be visible to other group members, the process should last seven days, the app should send push notifications with reminders to generate more ideas when the next phase starts and also if the last time the user opened the app was more than 24 hours ago. As the researchers want to find more about user decisions in the interviews after the session, they also activate tracking of submitted and rated ideas for all users.

Olivia is one of the experimentees. She downloads the app on her smartphone and enters her invitation code. After an introduction screen that explains the goal and procedure of the session, she can already submit her first idea. She types her idea and presses save. Now she sees her idea on the list. The next day she gets a notification about the beginning of the new phase. She notices an hour later because she studied and put her phone into silent mode. She clicks on the notification, the app opens, and she sees a notice informing her that now she can see, rate, and comment on the ideas from other group members. She sees all ideas from other participants and finds two exciting ones. She clicks on one of them and sees an overview with a description of an idea as well as an average rating and comments of other members. She rates the idea four stars and takes part in the discussion through comments.

Olivia has fun participating in the experiment but forgets to reopen the app if she does not become a reminder notification. On the last day of the experiment, she opens the app to check on the last results. She sorts the ideas by best average rating and is glad that an idea that she liked most was the second-best rated in the whole list.

After the experiment, the app shows the researchers user behavior data collected during the session. They export this data as a JSON file for further analysis in their favorite tools and proceed with conducting more experiments with another group and different settings and rules.

2.5 Use Cases

Use cases[JP15, p. 376] focus on user goals. They focus more on the user–system interaction rather than tasks. Use cases emphasize the interaction between the user (actor) and a system and focus mostly on the user's perspective rather than the system's. As the target audience of the Ideation-App consists of two different groups of users, the following are use cases for both kinds of interfaces.

Use Case - Experimenter's Perspective
1. The system displays input fields for an E-Mail and password.
2. The user enters their credentials.
3. The system checks if the credentials are correct.
4. The system displays a list of the projects and a button to add a new one.
4. The user chooses to add a new project.
5. The system displays input fields for project variables.
6. The user fills the fields.
7. The system checks if the input is valid.
8. The system displays the list of projects and a button to add a new one.
9. The user chooses one project.
10. The system displays project overview and options to export project data.
11. The user chooses to export the project data

Alternative courses:
4. If the credentials are incorrect:
— 4.1 The system shows an error message.
— 4.2 The system returns to step 1.
8. If the input is not valid:
— 8.1 The system shows an error message.
— 8.2 The system returns to step 5.

Use Case - Experimentees' Perspective
1. The system displays an input field for an invitation code.
2. The user enters an invitation code.
3. The system checks if the invitation code is correct.
4. The system displays a welcome page with a *next* button.
4. The user reads the text and presses the *next* button.
5. The system displays an option to add a new idea.
6. The user chooses to add a new idea.
7. The system displays input fields for a title and a description.
8. The user fills the fields.
9. The system checks if the title field is not empty.
10. The system displays a list of ideas.
11. The user chooses one idea.

10. The system displays an overview page of an idea: title, description, average rating, comments, and options to rate and comment on the idea.
11. The user chooses to rate an idea.
10. The system displays a rating dialog with four stars.
11. The user chooses five stars.
12. The system saves the rating and dismisses the rating dialog.
13. The user chooses to comment on the idea and presses on the input field.
12. The system shows a keyboard.
13. The user enters a comment.
14. The system checks if the comment is not an empty string.
15. The system adds the comment to the idea.
16. The user returns to the previous page.
17. The system displays a list of ideas.
18. The user chooses to sort ideas by number of comments.
19. The system displays a list of ideas sorted by number of comments.

Alternative courses:
4. If the invitation code is incorrect:
— 4.1 The system shows an error message.
— 4.2 the system returns to step 1.
10. If The title field is empty:
— 10.1 The system shows an error message.
— 10.2 The system returns to step 7.
15. If The comment is an empty string:
— 10.1 The system shows an error message.
— 15.2 The system returns to step 12.

2.6 Requirements

After collecting, sorting, and analyzing data gathered in the previous sections, the next step was establishing a list of requirements.

Table 2.2 displays the list of acquired requirements, grouped in three sections. The first section contains the requirements for admins' (experimenters') interface, the second for experimentees', and the last section displays requirements for both groups or general requirements for the app.

Table 2.2: Requirements

Admins should be able to ...	login to the admin interface with Email and password
	view a list of existing projects
	create new projects with different project variables (number of participants, start and end date and time, text for facilitation, content visibility options and start and end dates and times of phases, user consent options)
	see invitation codes of participants for each project
	export project data
Participants should be able to...	see the in-app guidance texts
	see how their data is used and stored
	give their consent to project admins to use and store their data
	see the phase description anytime
	see a list of ideas
	sort a list of ideas by date, average rating, number of ratings, number of comments
	create a new idea
	delete their own ideas
	view an average ratings of ideas
	rate ideas
	view comments of ideas
	comment on ideas
App should ...	run on mobile platforms (Android and iOS)
	run on smartphones and tablets
	be intuitive, easy to understand and to use
	have clean design
	work asynchronously
	offer secure authentication methods

2.7 App Mockups

Below are mockups divided into two groups, intended for two groups of stakeholders mentioned in the section 2.1. These mockups, created using Adobe Photoshop[9], are based on data gathered from interviews, analyzing similar software, and established requirements. The mockups served as blueprints for the design of the first low-fidelity prototype.

[9]https://www.adobe.com/products/photoshop.html

2.7.1 Experimenters' View

Figure 2.4 show mockups of the pages that the experimentees interact with.

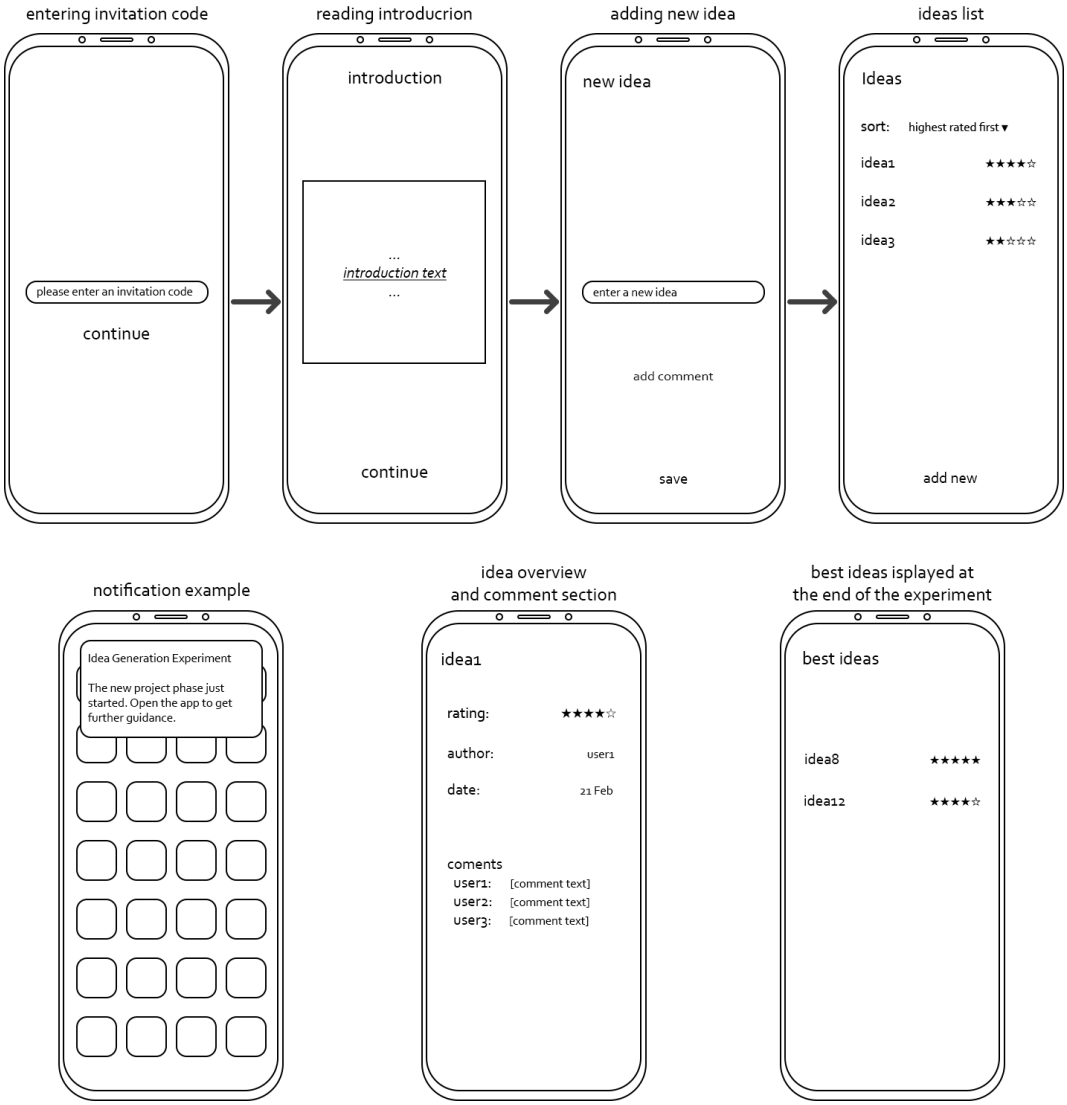


Figure 2.4: App Mockups - Experimentees' View

2.7.2 Experimentees' View

Figure 2.5 shows the mockups of the pages that are available only available for experimenters.

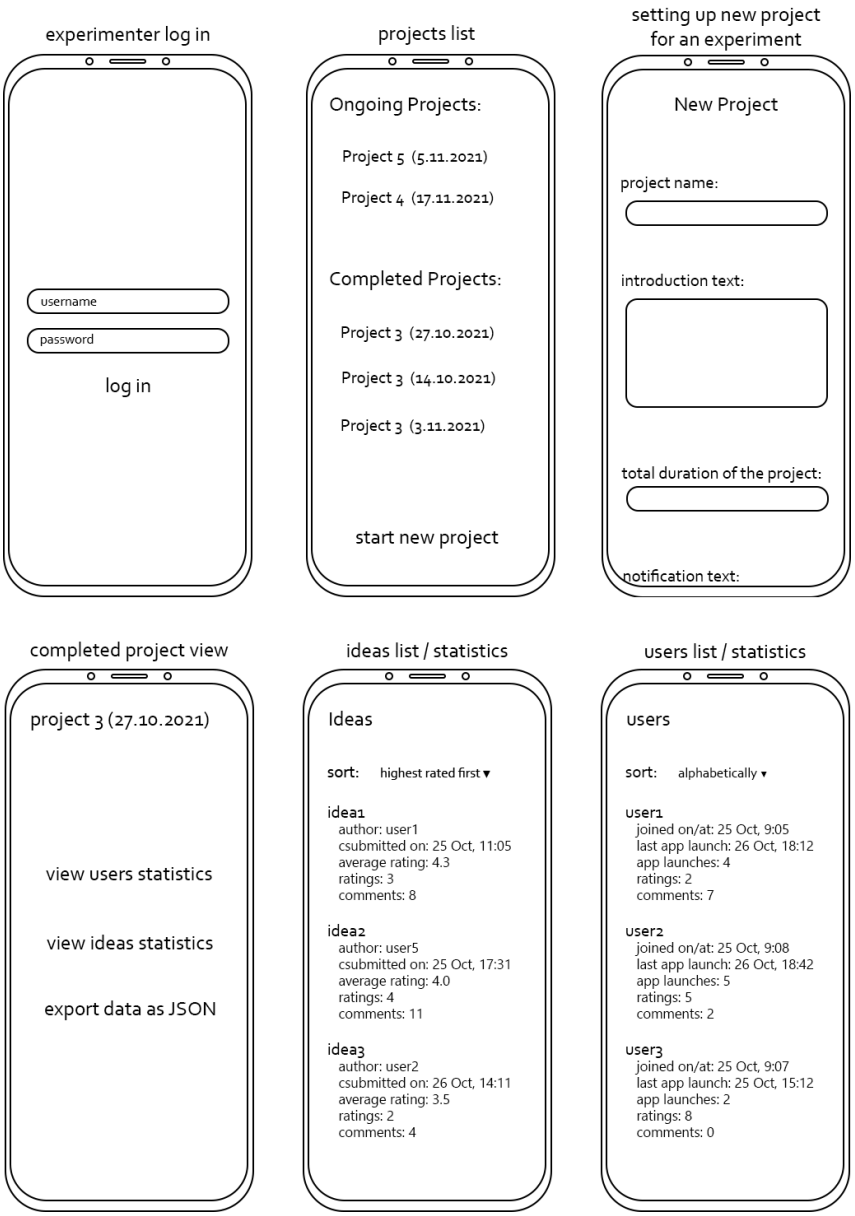


Figure 2.5: App Mockups - Experimenters' View

3 Implementation

3.1 Overview

The purpose of this chapter is to describe the implementation methods of the software prototypes, used tools, project structure, and its main components. The implementation phase had two main goals: First, implementing a low-fidelity prototype based on data gathered during the first design and data gathering phase. Second, implementation of a high-fidelity prototype that has enough functionality for field deployment and is based on data gathered while evaluating the low-fidelity prototype. The software project is available on GitHub[1].

3.2 Low-Fidelity Prototype

Flutter SDK[2] is often used as a rapid prototyping tool [Flu] as well as a production-level application development tool. As all user tests of the prototypes were going to proceed remotely during this work, in order to give participants some possibility of interaction with the software, the first prototype was implemented as a simple software prototype in Flutter.

Besides having a visual design based on mockups from Section 2.7, the prototype supported simple navigation through pages of the app and input fields for text. Almost all dynamic values of the application were static and served the sole purpose of being used in usability tests. As the usability tests were planned only from the perspective of experiment participants, only experimentees' screens (Figures 3.1 and 3.2) were prepared for the low-fidelity prototype.

[1] https://github.com/zurabtsc/ideation-app
[2] https://flutter.dev/

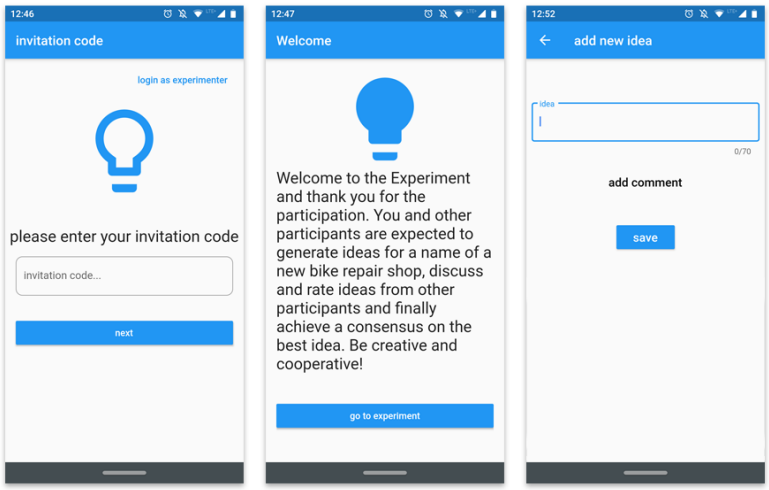


Figure 3.1: Low-Fidelity Prototype, Screens (set 1)

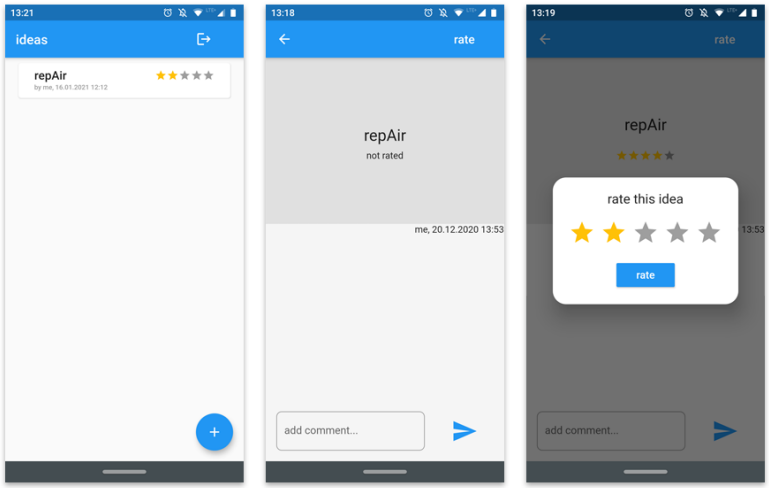


Figure 3.2: Low-Fidelity Prototype, Screens (set 2)

The prototype consists of five screens. The first screen that a participant of en experiment sees is an invitation code screen, which imitates user authentication with a valid invitation code. The following screen aims to welcome users and give them first information about the project through a short text. The middle screen on the Figure 3.1 contains an example of such text. Other screens imitate the basic purposes of the application: viewing, adding, rating, and commenting on ideas.

The goal of the implementation of the low-fidelity prototype was to prepare a testable model for usability tests. The evaluation of this prototype showed that completion of all given tasks was possible and that the prototype did not

have any notable bugs.

3.3 High-Fidelity Prototype

This section describes the implementation process of the high-fidelity prototype. First, there is a list of tools used for the implementation, and then descriptions of project structure, modules, screens, and visual elements. Finally, there is a short conclusion that sums up the progress made during the implementation.

3.3.1 Tools, APIs and Plugins

Following are the tools used during the implementation of the high-fidelity prototype:

- The software project is implemented in Flutter[3] SDK as a cross-platform application. Flutter is a free, open-source SDK that makes it possible to build apps for Android[4] and iOS[5] from a single codebase. It also supports necessary plugins for this project.
- Most of the code is written in Dart[6] programming language.
- Android Studio[7] served as an integrated development environment.
- For secure user authentication Firebase Auth[8] was implemented through Firebase Auth for Flutter v0.20.0 plugin[9].
- Online Database runs on Cloud Firestore[10]. The communication between the app and the server proceeds through the Cloud Firestore v0.16.0 Plugin[11].
- Offline persistent storage of key-value pairs is implemented through shared preferences v0.5.12 plugin[12] which uses NSUserDefaults[13] for iOS, and SharedPreferences[14] for Android based devices.

[3] https://flutter.dev/
[4] https://www.android.com/
[5] https://www.apple.com/ios
[6] https://dart.dev/
[7] https://developer.android.com/studio
[8] https://firebase.google.com/docs/auth
[9] https://pub.dev/packages/firebase_auth
[10] https://firebase.google.com/docs/firestore
[11] https://pub.dev/packages/cloud_firestore
[12] https://pub.dev/packages/shared_preferences
[13] https://developer.apple.com/documentation/foundation/nsuserdefaults
[14] https://developer.android.com/reference/android/content/SharedPreferences

- Interfaces for giving and viewing ratings use a configuration of Flutter rating bar v3.2.0 plugin[15].
- Notification system uses Flutter Local Notifications v4.0.1 plugin[16] for scheduling system notifications, which uses NotificationCompat[17] APIs for Android and UserNotifications[18] APIs for iOS.
- The application tests ran on Android Emulator[19] during the early stage of implementation, and later on an iPhone, Android smartphone, and Amazon Fire tablet.

3.3.2 Project Structure

The project consists of several Dart files that are thematically distributed into folders as follows:

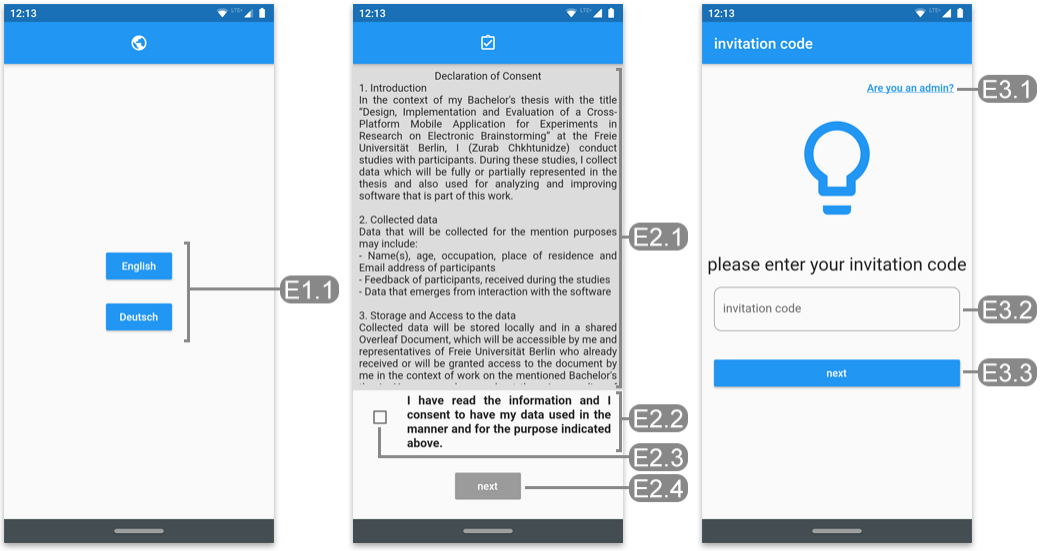
- **"APIs"** folder contains files that manage the connection between different parts of the app and the APIs used in the project.
- **"screens"** folder consists of files that represent all screens of the app in the form of Dart classes.
- **"components"** folder stores several UI elements that are either used repeatedly on different screens throughout the app or are too big to be part of the screen class file.
- **"styles"** folder contains files that store values for visual elements like colors or padding sizes.
- **"utils"** folder is for files that contain logic used throughout the whole project.

3.3.3 Application Screens

This section briefly describes application screens and their main elements. The screens are accessible either by experimenters or experimentees or by both of these groups. Start screens (listed in 5.1) are the first screens that users see when they open the app for the first time. Both groups see these screens. Experimenters' screens provide interfaces for the management of experiments and authentication of users with project management rights. On experimentees' screens, participants can use features for creating and interacting with the content of experiment projects.

[15] `https://pub.dev/packages/flutter_rating_bar`
[16] `https://pub.dev/packages/flutter_local_notifications`
[17] `https://developer.android.com/reference/androidx/core/app/NotificationCompat`
[18] `https://developer.apple.com/documentation/usernotifications`
[19] https://developer.android.com/studio/run/emulator

Figure 3.3: Start Screens



S1: Language Selection Screen

S2: Consent Screen

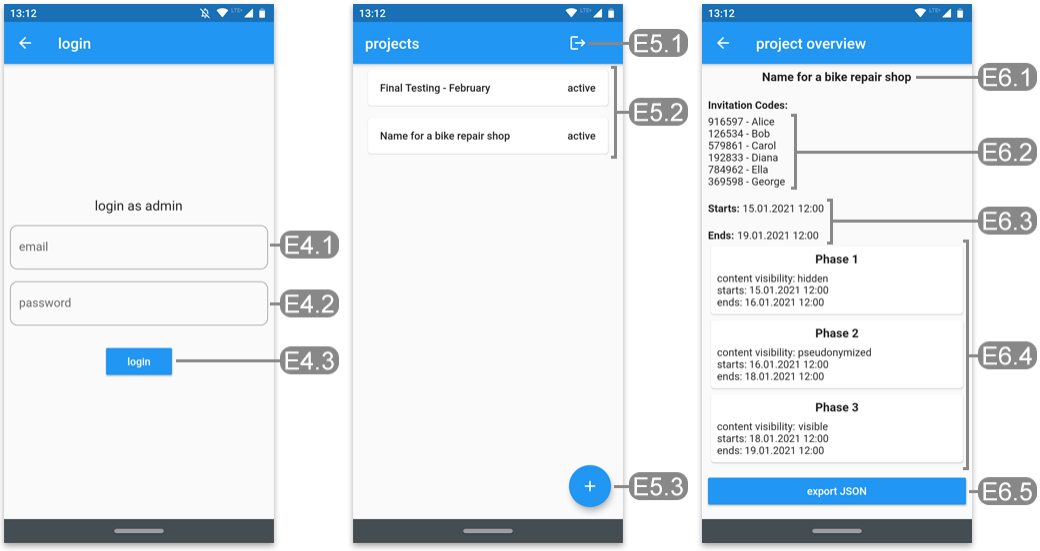
S3: Invitation Code Screen

Language Selection Screen (Figure 3.3, S1)
This is the first screen that users see when they launch the application for the first time. On this screen, users can choose options for the language of the app UI. After clicking on one of the buttons, the user navigates to the **Consent Screen** (S2), the language preference is saved locally and influences the appearances of other screens throughout the app as described in the Section 3.3.4.

Consent Screen (Figure 3.3, S2)
On this screen, users can read the declaration of consent (E2.1). The text appears in the selected language on Language Selection Screen(S1). To agree to the conditions, users have to check the checkbox (E2.3) and press the "next" button (E2.4), which is only active when the checkbox is checked. Initially, the checkbox is unchecked. After pressing the active "next" button, users navigate to **Invitation Code Screen** (S3).

Invitation Code Screen (Figure 3.3, S3)
Experiment participants can enter their invitation codes on this screen into the input field (E3.2). After entering the invitation code and clicking on the "next" button (E3.3), the system performs online verification of the correctness of the code and, in case of confirmation, navigates the user to the *Project Welcome Screen* (S8). For authentication of admins (or experimenters), there is a button (E.3.1) that leads to *Admin Login Screen* (S4).

Figure 3.4: Experimenters' Screens - Set 1



S4: Admin Login Screen

S5: Projects List Screen

S6: Project Overview Screen

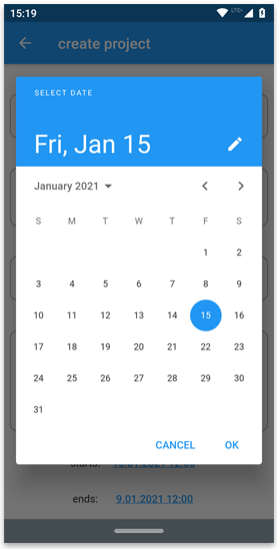
Admin Login Screen (Figure 3.4, S4)
Experimenters can authenticate themselves on this screen with their personal login data. After entering E-Mail and Password in the input fields (E4.1, E4.2) and pressing the "login" button, the system tries to verify the authenticity of the entered data and, upon success, navigates the user to *Projects List Screen* (S5).

Projects List Screen (Figure 3.4, S5) This screen displays a list of existing projects (E5.2) as cards, each with project title and status. All cards are clickable and lead to *Projects Overview Screen* (S6) with details of the selected project. Administrators can create new projects by clicking on the "+" floating button (E5.3). In this case, the app navigates to *Project Creation Screen* (S7.1). By pressing on the "log out" icon button (E5.1), the system dismisses locally saved authentication data, and the user navigates back to *Admin Login Screen* (S4).

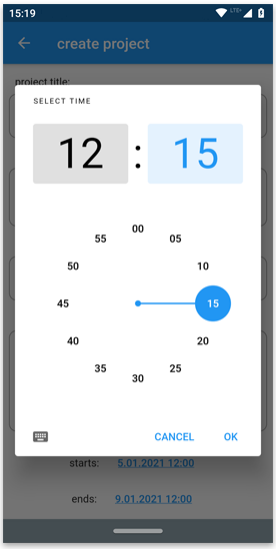
Projects Overview Screen (Figure 3.4, S6) On this screen, experimenters can view information about the selected project and export all collected data. The following information is available on this screen: project title (E6.1), invitation codes of participants (E6.2), start and end date and time points of the project (E6.3), and information about all phases (E6.4). After pressing on "export JSON" button (E6.5), admins will be presented with the options to export all project-related data in JSON format.

Figure 3.5: Experimenters' Screens - Set 2

S7.1: Project Creation Screen



S7.2: Project Creation Screen - Date Picker



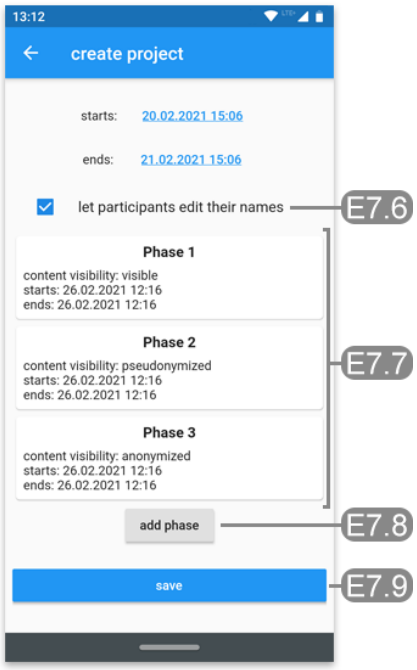
S7.3: Project Creation Screen - Time Picker

Project Creation Screen (Figure 3.5)
Admins can create new experiment projects on this screen. Here they can set any of the values of project variables. This screen is implemented as a scrollable [Scr] so that potentially any number of input fields can fit the view.

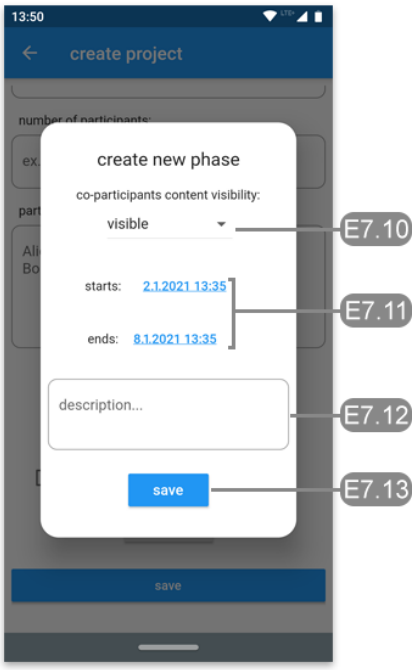
Element E7.1 is an input field for a name of a new project. This name is visible on the **Projects List Screen (S5)**. Welcome text (E7.2) appears on the **welcome screen (S8)**. This is the first text message that participants see after entering their invitation codes. Therefore, experimenters can use this text to introduce the project and describe the first steps that users should take in an experiment.

The value in the element E7.3 defines the number of invitation codes that are available for experimenters after creating a new project. Admins can also enter some or all names of participants into the next input field (E7.4). Certainly, the number of these names should not be greater than the number entered in E7.3. The links of the element E7.5 provide functions for setting the start and the end time points of the project. This action leads to opening a date picker popup (S7.2) where admins can select the date and then proceed to the time picker popup (S7.3) to select the time and confirm the selection.

Figure 3.6: Experimenters' Screens - Set 3



S7.4: Project Creation Screen



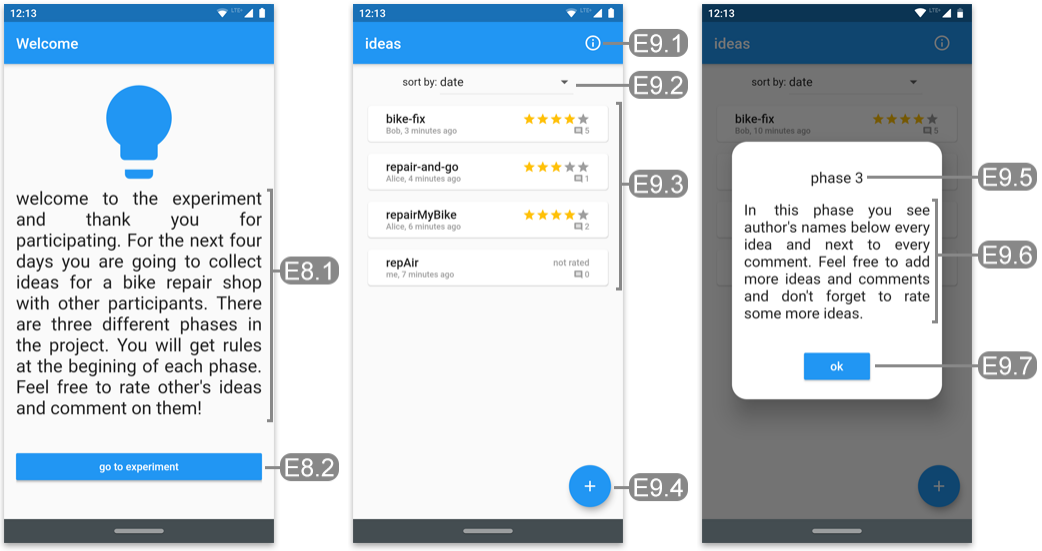
S7.5: Project Creation Screen - Phase Popup

Project Creation Screen (Figure 3.6, S7.4) The screen S7.4 displays more elements of the list presented on *Project Creation Screen* (S7.1). During creating a new experiment, experimenters can let participants edit their names by setting a checkmark in E7.6.

Admins can add new phases of the experiment by pressing a *add phase button*. This action causes the *create new phase popup* (S7.5) to appear. After pressing the *save button* (E7.9), the system saves the project data, synchronizes it with the cloud, and navigates the user back to *Projects List Screen* (S5).

Phase Popup (Figure 3.6, S7.5) On this popup, users can choose options for a new phase of the project. It is a dialog where admins can set content visibility options (E.10), start and end time points of the phase (E.11), and a description (E7.12). After saving a new phase by pressing the *save button* (E7.13), the newly added phase appears in the block of all previously created phases E7.7.

Figure 3.7: Experimentees' Screens - Set 1



S8: Welcome Screen

S9.1: Ideas List Screen

S9.2: Ideas List Screen - Info Popup

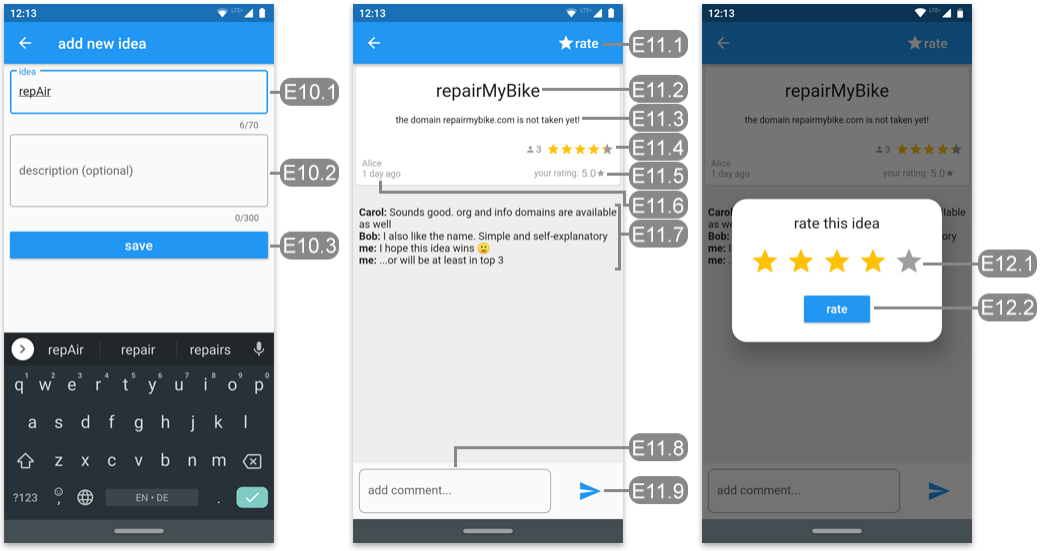
Project Welcome Screen (Figure 3.7, S8)
This screen provides the first information about the experiment to participants. The text displayed on this screen is variable, depending on the input on the *Project Creation Screen* (S7.1) in the "welcome text" field (E7.2). After pressing the "go to experiment" button(E8.2), users navigate to *Ideas List Screen* (S9.1).

Ideas List Screen (Figure 3.7, S9.1 and S9.2)
On this screen, participants see a list of ideas (E9.3) that they or (depending on the content visibility settings of the current phase of experiment) other participants committed to the project so far. The drop-down menu (E9.2) offers an option to sort the list of ideas by different properties.

A representation of each idea is a card with an idea title, author's name, creation time, average rating, and the number of comments. The author's name field varies depending on the following factors: if a participant views their own idea, the field's value is always "me". Else, depending on the content visibility of the phase (open, pseudonymized, anonymized), the visible value is the author's real name, their pseudonym, or a label 'anonymous' respectively. Clicking on an idea card navigates users to the *Idea Overview Screen* (S11), where they can rate or comment on this idea and view more information about it.

The *info button* (E9.1) opens the *info button* (E9.1) with the description of the current phase. The floating "+" button (E9.2) leads to the *Idea Creation Screen* (S10).

Figure 3.8: Experimentees' Screens - Set 2



S10: Idea Creation Screen

S11: Idea Overview Screen

S12: Idea Overview Screen - rate popup

Idea Creation Screen (Figure 3.8, S10)
Experimentees can commit new ideas from this screen. The first input field (E10.1) is for a title or a short form of an idea. This field has a character limitation of 70. The second input field (E10.2) is for an optional description of the idea. A click on the *save button* (E10.3) results in sending the idea to the database, and navigation of the participant back to the *Ideas List Screen* (S9.1), where the newly committed idea appears in the *ideas list* (E9.3).

Idea Overview Screen (Figure 3.8, S11)
This screen displays the idea, selected on the *Ideas List Screen* (S9.1) and all available, related information to it. Users can see the title (E11.2) of the idea, the description (E11.3), their rating of the idea (E11.5), average rating and a number of users who rated this idea (E11.4), author and creation time (E11.6), and the comments list (E11.7). Users can add their comments by entering them in the *comment input field* (E11.8) and pressing the *send button* (E11.9). They can also add or edit their rating of this idea by pressing either E11.1 or E11.4, which opens a *rate popup* (S12). Here a rating can be given by selecting stars (1-5) (E12.1) and pressing the *rate button* (E12.2), which saves the rating and dismisses the popup. If a user decides to edit their rating, the system sends the new one to the database, but old ones will remain in the database since this data might be valuable for later analysis. Therefore, the average rating of the idea is calculated from the sum of all current user-unique ratings divided by their number.

3.3.4 Language Management

The software supports different languages. During the evaluation, English and German languages were available in the app. The project can potentially support any number of languages.

On the first screen that appears after users launch the app for the first time, they can choose one of the available languages. After that, a variable holding the language code of the selected language is stored in persistent memory through shared preferences.

The map of string translations is stored in *localization.dart* file in *utils* folder. Every entry in the map has a unique key (mostly a short variant of the English string) and a map as a value, which holds language codes as keys and translated strings as values. The system accesses the strings through the *localStr(String key)* function with a unique key as an argument. The function returns the requested string for the language code saved in shared preferences.

3.3.5 Consent management

After choosing the language of the app, the user navigates to the consent screen. On this screen, users can view the information about experiments, data usage, conditions of participation, and more. All strings on this page appear in the language that the user chose on the previous page. These strings are configurable by admins.

Users can agree to the conditions and consent for data usage by putting a checkmark into the displayed checkbox at the bottom and pressing the "next" button. The checkbox is initially unchecked. The "next" button does not invoke any functions, so the user can only proceed to further screens if they grant the consent.

3.3.6 User Management

The app manages two user groups: admins (experimenters), who create and administrate experiment projects, and participants (experimentees), who take part in experiments.

Access to the administrator interface is possible through E-Mail and password authentication provided by Firebase Auth. Creating new administrator accounts is possible through Firebase Console.

After researchers choose settings for a new project, the app generates invitation codes for participants. The number of codes depends on the number of participants defined in the project settings. Experimentees must install the app on their smartphones and enter an invitation code. Then they can start participating in an experiment.

The participant interface is accessible by entering an invitation code on the invite code page. After entering a code, the system checks which project is assigned to the entered code and saves it in the shared preferences. On the

next launch of the app, the system checks for saved codes, and if it finds a valid one, the user automatically navigates to the participant's interface.

3.3.7 Facilitation

The application provides several means of facilitating the ideation process. The first experiment-related communication can take part on the *Project Welcome Screen* (Figure 3.7, S8). At the start of every new phase, users see a popup with a description of the phase determined on *Project Creation Screen* (Figure 3.6, S7.4) by admins. Another opportunity for facilitation of the experiment is the notification system. Contents for all these elements are variable and open to experimenters for conducting research studies with different conditions.

3.3.8 Data Management

There are three main data storage mechanisms implemented in this project:

- online database used for controlling the user authentication managed through Firebase Auth,
- online database for storing and retrieving project-related data for every project,
- and an offline key-value storage mechanism managed through shared prefs for saving information on the device after the app is closed.

Almost all data related to experiment projects is stored online, in Cloud Firestore in a NoSQL database. These databases consist of collections of documents. For every new experiment project, a new collection is created, which contains all ideas as documents. Idea documents contain information about ideas, such as creation time point, author, all ratings, and comments. There are two more collections in the database that hold general data for all experiment projects: *_app_data* collection holds configuration values like counters for project IDs, *_projects_data* collection contains documents for each experiment project with information about projects including creation time point, description, phases, names and invitation codes of participants and a counter for idea IDs.

3.3.9 Project Management

Admins can create experiment projects, manage them, and see the list of all projects and detailed information about any of them.

On the project creation screen, admins can choose between several different settings as seen in the Figure 3.5, S7.1.

3.3.10 Rating System

Users can rate and re-rate other users' ideas through the 5-star rating system. All user ratings are stored in the online database. When participants view ideas, the system calculates the average ratings and displays them. During calculations of an average rating, only the most recent ratings for each user are considered.

3.3.11 Commenting System

Participants of experiments can comment on their own or other participants' ideas. The comments list looks like a chat and is visible in the idea Overview screen under the details of the idea. Users see their own comments labeled with the string "me". Depending on content visibility settings of the current phase, other participant's comments are labeled with author's name when content visibility is set to visible, with their pseudonym when pseudonymized, or with a string "anonymous" when content is anonymized. If the content visibility of a phase is set to hidden, users see only their own comments.

3.3.12 Notifications System

System notifications remind users to open the app and commit more to experiment at the scheduled time points. During the configuration of a new project, administrators can choose if the notifications should show after the end of a project phase and also at certain time points.

After clicking on a notification tile, the app launches, and users navigate to the Ideas List Screen.

3.3.13 Collecting and Exporting Data

Administrators can export data of any project in JSON Format during or after any experiment. Researchers can use these JSON files in other software for analyzing or visualizing experiment data. Exported data contains all the stored information about projects, participants, ideas, comments, and ratings.

3.3.14 Conclusion

The goals of this section were to implement prototypes that would be suitable for user tests. The evaluation of the low-fidelity prototype did not reveal any significant implementation-related problem. In the evaluation phase of the high-fidelity prototype, some minor problems regarding the implementation were discovered (described in the Section 4.2.3). However, participants could still complete their tasks without notable technical problems.

4 Evaluation

This chapter describes the process and results of evaluating developed prototypes during this work. First, there is a discussion of the conducted usability tests of the low-fidelity prototype [JP15, p. 389-391]. Then, there is a review of the preparation and execution of field deployment [SHNT14] of the high-fidelity prototype [JP15, p. 391-392] and the results of the evaluation.

4.1 Usability Tests

The goal of usability tests [JP15, p. 457-459] was to identify possible issues in the low-fidelity prototype and gather feedback and improvement suggestions.

Two participants (P4 and P5, Appendix 5.3) took part in software testing sessions to evaluate the prototype. These sessions had a form of remote usability tests [JP15, p. 478]. These tests were held online, through a video call and screen sharing. Participants received consent forms before the tests. The next step was to prepare questions and the protocol. At the beginning of the tests, the participants received information about the purpose and process of the tests as well as the purposes of data usage and storage.

Before proceeding to the actual test, it was made clear to participants that the prototype and not them are being tested. The first phase of the test utilized the think-aloud technique [JP15, p. 260-262]. Participants received four tasks to complete in the prototype:

1. log in using given invitation code,
2. create a new idea,
3. rate an idea,
4. write a comment.

At every step, participants spoke their thought and expectations aloud. They left their feedback on every screen and most elements of the app. Testing the prototype also helped to identify numerous problems, and in some cases, the participants proposed their solutions.

The second phase was the feedback phase. Participants expressed their general thoughts on the app, sections, and components that, in their opinion, need improvement.

The next step included analysis of the transcripts of participants' responses (Appendices 5.7 and 5.8). The list below summarizes the results of the usability tests of the prototype.

Suggestions and comments made by two participants:

- texts for app guidance should be available for reviewing anytime in the app
- "add comment" button does not look like a clickable element
- number of ratings and number of comments should be visible for every idea on the ideas list
- users should be able to sort or group ideas
- initial value in rating popup should not be three; a better value would be zero
- ratings should be given only in whole numbers
- the participants must be able to edit and delete their ideas
- after clicking on system notification, the app should open the ideas list screen

Suggestions and comments made by only one participant:

- button label "login as experimenter" is somewhat confusing
- better explanation to add an idea is needed when the ideas list is empty
- when adding ideas, the first comment should be replaced by a description
- font size of secondary texts on ideas cards is too small rating comments might be helpful
- participants should be able to see the list of all participants
- there should be some sign that indicates that a participant has already rated an idea
- list of recent activities should be visible in the app
- users must be able to subscribe to new activities and receive notifications with updates
- the system back button should always navigate the user to the previous screen

The usability tests of the low-fidelity prototype were to identify possible issues in the prototype and gather feedback and improvement suggestions. Identification of multiple points of improvement adjusted the future direction of the project. These suggestions mostly formed the design and implementation of the high-fidelity prototype.

4.2 Field Deployment

This section discusses the evaluation of the high-fidelity prototype. First, there is a description of the planning of the experiment with the necessary steps. Then, there is a review of the preparation and execution, and finally, the experiment results. The goals of the field deployment were to find out if the features of the application function as expected in a natural setting as well as to measure the usability of the app and get feedback and improvement suggestions from participants.

4.2.1 Planning

As the app's implementation reached the state where it could be tested "in the wild", in a setting that is close to the real expected environment, preparations for the field deployment started.

Finding participants and deciding on the goal and duration of the field deployment was the first planned activity. Afterward, the project with the planned number of participants, duration, guiding texts, consent forms, and other relevant variables had to be prepared.

E-Mails with invitation codes and technical instructions had to be sent to participants, so they had all the necessary information to start participating in the experiment at the planned time.

Online surveys had to be prepared on a convenient online platform, and experimentees had to be asked to participate in them after the experiment.

After the experiment and completion of the survey phase, user activity data from the app and results of the surveys had to be extracted, analyzed and some of the data had to be visualized.

4.2.2 Preparation and Execution

Six members of a students' association in Bochum (P6-P11, Appendix 5.3) agreed to participate in the field deployment. As they plan to develop an app for their association soon, they are looking for a name for the app, so the experiment's goal was to find a name for the planned app.

After the preparation of the project in the app, it had new values. The duration of the project was four days. This period consisted of three phases, each with different content visibility options:

- hidden - 24 hours

- pseudonymized - 48 hours
- open - 24 hours.

For this evaluation, the app used the text of the consent form from the previous tests on a consent form page.

Participants received their invitation codes generated by the app via E-Mail with a short description of the next steps.

Google Forms[1] served as a platform for conducted surveys after tests. The first questions were about participants, their names, age, and occupation. Then, to measure the user experience of the app, the German version of questions of UEQ[2] was integrated. In the next part, the survey had questions about the usage of similar apps (Google Keep, Google Docs, WhatsApp). In the end, there were open questions asking participants what they liked about the app and what could be improved.

During the field deployment, two minor bugs were captured: first, one attempt to rate an idea resulted in several identical data entries in the database; second, the app's icon did not appear on some devices next to the notification message. The identical entries were cleared from the database, so the validity of the whole data set was not threatened.

Data from the app was exported in JSON format[3]. It contains information about all participants, ideas, comments and ratings, and general information about the project itself. Some data parts were distributed to several files in CSV format[4] and later imported into a Google Spreadsheet[5] where they were visualized.

Data from the surveys was extracted from Google Forms and was analyzed. Answers to the questions from UEQ were analyzed and visualized in the UEQ Data Analysis Tool [Sch19, p. 5-7].

4.2.3 Results

The experiment data originated from two sources: the tracking system of the app and Google Forms. The app's tracking system provides options to export all experiment data in JSON format. Results from online surveys were exported and were later analyzed and visualized with the UEQ Data Analysis Tool.

The survey showed that all participants use WhatsApp daily, Google Docs two to three times a week, and Google Keep about once a week on average. These responses indicate that participants were familiar with similar concepts and features and that they used them frequently. This might have made it easier for them to understand the user interface and features of the app. Useful

[1] https://www.google.com/forms/about/
[2] https://www.ueq-online.org/
[3] https://www.json.org/json-en.html
[4] https://www.loc.gov/preservation/digital/formats/fdd/fdd000323.shtml
[5] https://www.google.com/sheets/about/

information may be found by conducting a similar experiment with participants who have less experience with similar apps.

As answers to the last two open questions of the survey, participants wrote about parts that they liked in the app and also about parts that, in their opinion, could be improved. On average, participants found the app somewhat helpful and intuitive. They responded positively to the existence of phases with different rules. The most suggested improvement was a feature that sends more notifications for (almost) every new activity in the app. One user found the visual design too minimalistic and suggested adding more visuals. Other user pointed out the lack of a screen where a summary of the final results of the experiment could be displayed.

Exported usage data from the app showed that the main features of the app were usable. Every participant added and rated ideas. Five out of six participants also wrote comments.

Some visualizations in this section demonstrate the possible use of collected app usage data. For example, figure 4.1 shows how many ideas, ratings, and comments were committed by every of six participants.

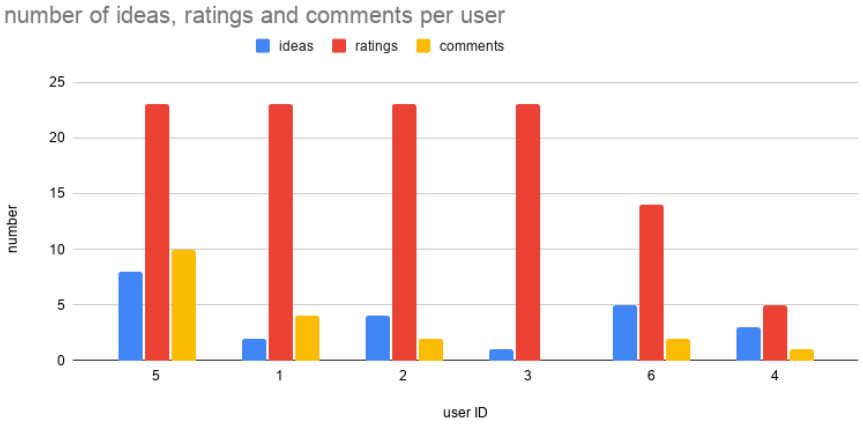


Figure 4.1: Number of ideas, ratings and comments per user

Figure 4.2 demonstrates average rating of every idea. The average rating of the best-rated idea is 3.33. This suggests that participants rated ideas rather critically.

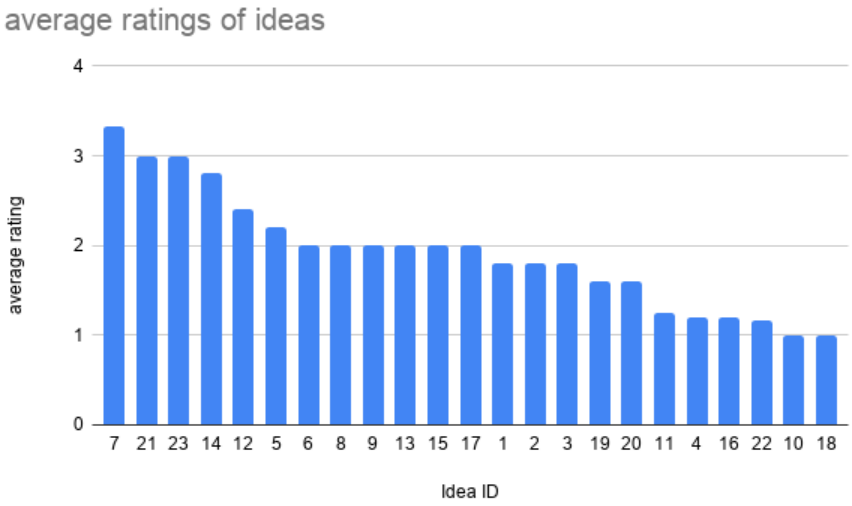


Figure 4.2: Average ratings of ideas

Figure 4.3 shows how many comments and ratings were committed per idea. As shown, every idea was rated by at least four participants, but less than half of them received comments. One can conclude that users were more motivated to rate ideas than to comment on them. Conducting more experiments with different techniques to motivate users to comment more on ideas may show more interesting results.

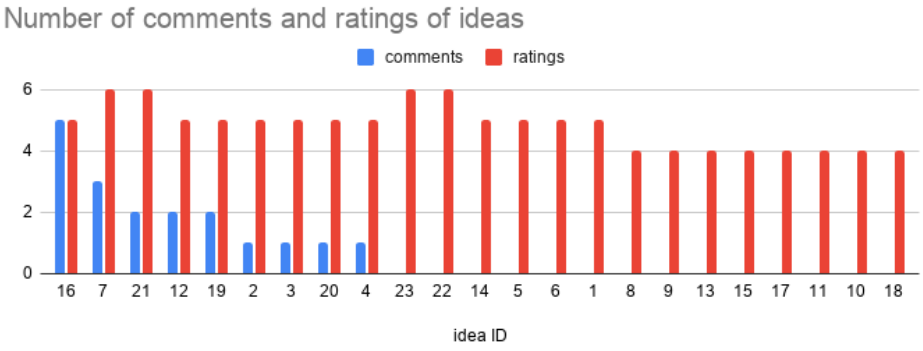


Figure 4.3: Number of comments and ratings of ideas

In the figure 4.3 the timeline of the whole period of the experiment is presented. Two markers for notifications indicate the times when every participant received a reminder notification. First Notification was 24 hours after the start of the experiment. As seen, big activity was registered shortly after this notification. The second notification was sent 72 hours after the start of the project and was followed by some ratings of ideas from participants.

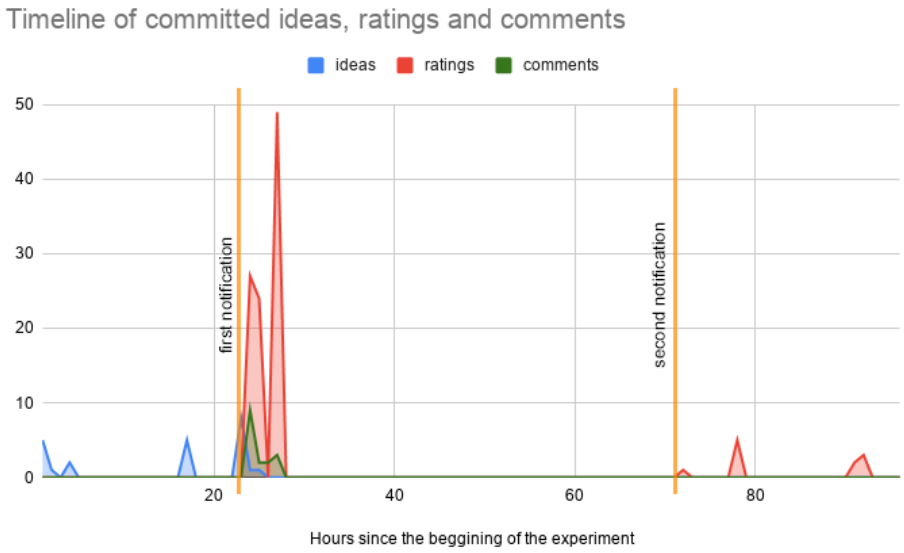


Figure 4.4: Timeline of notifications and numbers of committed ideas, ratings and comments

The data from surveys was analyzed and visualized in UEQ Data Analysis Tool. This tool calculated a benchmark (shown on the Figure 4.5) for this app. As shown, the lowest score was given in the category of dependability. This category measures if the user feels in control of the interaction and whether the product is secure and predictable.

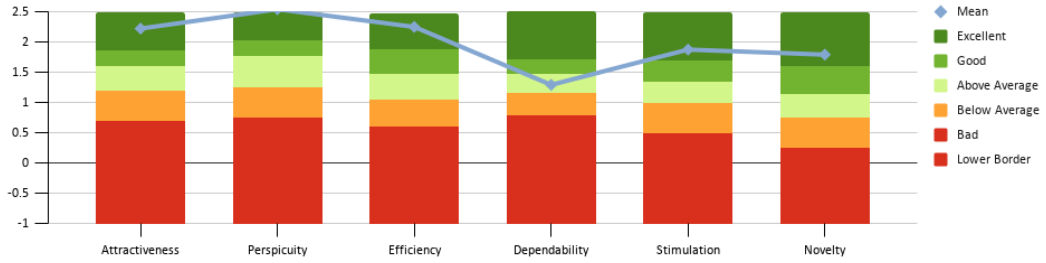


Figure 4.5: UEQ Benchmark

Following the goal to test the app features, the results showed that the application delivered the expected functionality altogether, although some bugs were found and identified. After measuring the app's usability and calculation of the benchmark, participants submitted their feedback and improvement suggestions. As the number of participants was rather low, the evaluation may have missed important findings through surveys or usage data. Due to the same reason, the benchmark may show imprecise values. Evaluating such an experiment with more participants could deliver more important information.

5 Conclusion

The main goal of this work was to develop a prototype of a cross-platform mobile application for deployment in experimental studies on electronic brainstorming with the help of mobile devices focusing on the experimentees' interface. The evaluation of the final prototype of this work helped to identify some points of improvement. Nevertheless, the participants of field deployment were mainly able to understand the interface effortlessly and accomplish the tasks using all features of the app.

The second goal was to work on the prototype focusing on usability in cycles of designing, implementing, and evaluating. After gathering the first data through semi-structured interviews and the analysis of similar software, the first prototype was designed and implemented. Usability tests helped to evaluate the prototype. The design of the high-fidelity prototype emerged from the gathered data from the usability tests. The tests of the high-fidelity prototype proceeded 'in the wild'. The analysis of these tests resulted in the extraction of valuable data for further development.

In future work, several new features might be worth implementing. Displaying a screen with a summary of results shortly before completing an experiment and giving users an opportunity to make their last changes and contributions could lead to changes in the final results. Thus this feature could be interesting for researchers to explore such user behavior. Another helpful feature could be a kind of notifications management system that is open for participants, where they can subscribe to more app events, such as new comments or new ideas.

As this work mostly focuses on participants' interfaces, the experimenters' interfaces do not offer some comfort features, such as displaying more detailed statistics of the projects and visualizing the results directly in the app. Another useful feature might be sending invitation codes automatically and directly from the app to participants E-Mail addresses and, generally, communicating to participants through E-Mail before, during, and after an experiment. Such features might be helpful for researchers and might be worth designing, implement and evaluate.

During the work, participants of the field deployment and interviews significantly contributed to the development of the software through their diligent cooperation and valuable feedback. The topic of the field deployment was interesting for the participants. Involving larger groups would possibly lead to capturing more and different issues. Possibly the biggest issue with remote testing was that the problems that are identifiable only during direct observation remained unnoticed. Another possible issue might be that the problem that participants tried to solve during the field deployment was not of greater significance, which would possibly lead to more user activity and thus to gen-

erating more data for identifying issues. Conducting interviews and deploying field studies with larger audiences, and choosing topics of greater significance may lead to finding more points of improvement and interesting potential features in future work.

Bibliography

[APAa] APA Dictionary of Psychology: brainstorming. https://dictionary.apa.org/brainstorming, last accessed on 20.03.2021.

[apab] APA Dictionary of Psychology: electronic brainstorming. https://dictionary.apa.org/electronic-brainstorming, last accessed on 20.03.2021.

[Bra] How to use brainstorming effectively. https://community.hrdaily.com.au/profiles/blogs/how-to-use-brainstorming-effectively, last accessed on 20.03.2021.

[Car00] J.M Carroll. Introduction to this special issue on "scenario-based system development". *Interacting with Computers*, 13:41–42, 2000.

[CDD16] Joel Chan, Steven Dang, and Steven P. Dow. Ideagens: Enabling expert facilitation of crowd brainstorming. In *Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion*, CSCW '16 Companion, page 13–16, New York, NY, USA, 2016. Association for Computing Machinery.

[Flu] Flutter prototyping for rapid mobile application development. https://www.star-knowledge.com/blog/flutter-prototyping-for-rapid-mobile-application-development/, last accessed on 20.03.2021.

[GK98] Ian Sommerville Gerald Kotonya. *Requirements Engineering: Processes and Techniques*. Wiley, 1998.

[HCD] ISO 9241-210:2019 Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems. https://www.iso.org/standard/77520.html, last accessed on 20.03.2021.

[HCD08] Thomas Howard, S.J. Culley, and Elies Dekoninck. Describing the creative design process by the integration of engineering design and cognitive psychology literature. *Design Studies*, 29:160–180, 03 2008.

[IDD18] Nanna Inie, Peter Dalsgaard, and Steven DOW. Designing idea management tools: three challenges. 06 2018.

[ide] APA Dictionary of Psychology: brainstorming. https://dictionary.apa.org/ideation, last accessed on 20.03.2021.

[JP15] Yvonne Rogers Jennifer Preece, Helen Sharp. *Interaction Design: Beyond Human-Computer Interaction*. Wiley, 4th edition, 2015.

[Lub05] Todd Lubart. How can computers be partners in the creative process: Classification and commentary on the special issue. *International Journal of Human-Computer Studies*, 63:365–369, 10 2005.

[Mat] Material Design. https://material.io/design, last accessed on 20.03.2021.

[MIT] MIT sloanreview: brainstorming electronically. https://sloanreview.mit.edu/article/brainstorming-electronically/, last accessed on 20.03.2021.

[Osb53] Alex Faickney Osborn. *Applied Imagination: Principles and Procedures of Creative*. New York, Scribner, 1953.

[PBWK19] Leonard Przybilla, Luka Baar, Manuel Wiesche, and Helmut Krcmar. Machines as teammates in creative teams: Digital facilitation of the dual pathways to creativity. pages 94–102, 06 2019.

[Sch19] Dr. Martin Schrepp. *User Experience Questionnaire Handbook*. Team UEQ, 8th edition, 2019.

[Scr] Scrollable Class, Flutter. https://api.flutter.dev/flutter/widgets/Scrollable-class.html, last accessed on 20.03.2021.

[SHNT14] Katie A. Siek, Gillian R. Hayes, Mark W. Newman, and John C. Tang. *Field Deployments: Knowing from Using in Context*, pages 119–142. Springer New York, New York, NY, 2014.

Appendix

5.1 Declaration of Consent (English Version)

Declaration of Consent

1. Introduction

In the context of my Bachelor's thesis with the title “Design, Implementation and Evaluation of a Cross-Platform Mobile Application for Experiments in Research on Electronic Brainstorming” at the Freie Universität Berlin, I (Zurab Chkhtunidze) conduct studies with participants. During these studies, I collect data which will be fully or partially represented in the thesis and also used for analyzing and improving software that is part of this work.

2. Collected data

Data that will be collected for the mention purposes may include:

- Name(s), age, occupation, place of residence and Email address of participants
- Feedback of participants, received during the studies
- Data that emerges from interaction with the software

3. Storage and Access to the data

Collected data will be stored locally and in a shared Overleaf Document, which will be accessible by me and representatives of Freie Universität Berlin who already received or will be granted access to the document by me in the context of work on the mentioned Bachelor's thesis. You can read more about the privacy policy of Overleaf on https://www.overleaf.com/legal

Data that will be represented in the Bachelor's thesis and will be accessible by everyone who has access to the mentioned Bachelor's thesis. All other data will be deleted no later than 14 days after the final presentation date of the thesis.

4. Contact

For requests and questions about the collected data please contact me via Email: zurab.tsc@gmail.com

consent

I have read the information, or it has been read to me. I have had the opportunity to ask questions about it and my questions have been answered to my satisfaction. I consent voluntarily to have my data stored in the manner and for the purpose indicated above.

Print Name of Participant _____

_____ Date (Day / month / year)

_____ Signature of Participant

5.2 Declaration of Consent (German Version)

Einverständniserklärung

1. Einleitung

Im Kontext meiner Bachelorarbeit mit dem Titel “Design, Implementation and Evaluation of a Cross-Platform Mobile Application for Experiments in Research on Electronic Brainstorming” an der Freien Universität Berlin führe ich (Zurab Chkhtunidze) Studien mit Teilnehmenden durch. Während diesen Studien sammle ich Daten, die vollständig oder teilweise in meiner Bachelorarbeit repräsentiert und für die Verbesserung der im Rahmen dieser Arbeit zu entwickelten Software benutzt werden.

2. Gesammelte Daten

Daten, die für oben genannten Zwecken gesammelt werden, können folgendes enthalten:

- Name(n), Alter, Beruf, und E-Mail Adresse der Teilnehmenden
- Rückmeldung der Teilnehmenden, die während der Studien eingesammelt wird
- Daten, die durch Interaktion mit der Software erstellt werden

3. Speicherung und Zugriff auf die Daten

Gesammelte Daten werden sowohl lokal als auch in einem geteilten Overleaf Dokument gespeichert. Den Zugriff auf dieses Dokument werden, außer mir, Angehörige der Freien Universität Berlin haben, denen ich im Kontext dieser Bachelorarbeit Zugang gewährt habe oder diesen gewähren werde. Über Datenschutz-Bestimmungen von Overleaf können Sie mehr unter dem folgenden Link erfahren: https://www.overleaf.com/legal

Daten werden außerdem in der Bachelorarbeit repräsentiert und werden daher für alle zugänglich sein, die Zugang zu der oben genannten Bachelorarbeit haben. Alle weitere Daten, die in der Bachelorarbeit nicht repräsentiert werden, werden spätestens 14 Tage nach der Präsentation der Arbeit gelöscht.

4. Kontakt

Für Anfragen über gesammelten Daten können sie mich über folgende E-Mail Adresse kontaktieren: zurab.tsc@gmail.com

Erklärung

Ich habe die Information gelesen oder sie wurden mir vorgelesen. Ich hatte die Möglichkeit Fragen zu stellen und meine Fragen wurden zu meiner Zufriedenheit beantwortet. Ich bin damit einverstanden, dass meine Daten in der oben angegebenen Weise und zu dem oben angegebenen Zweck gespeichert und verwendet werden.

Name des Teilnehmers / der Teilnehmerin: _____

_____ _____

Datum (Tag / Monat / Jahr) Unterschrift des Teilnehmers / der Teilnehmerin

5.3 Participants

P1: 27 years old game system design student (M.A.) from Berlin.

P2: 58 years old lead of engineering at a software development company from Bielefeld.

P3: 38 years old film director and university lecturer from Berlin.

P4: 26 years old computer science student (M.Sc.) from Berlin.

P5: 21 years old economics student, B.Sc. from Bielefeld.

P6: 29 years old law student (LL.B.) from Bielefeld.

P7: 26 years old biology student (B.Sc.) from Bochum.

P8: 28 years old business management student (B.Sc.) from Bochum.

P9: 25 years old event management and economics student (dual course of studies, B.Sc.) from Bochum.

P10: 29 years old business information systems student (B.Sc.) from Hamburg.

P11: 22 years old biology student (B.Sc.) from Bochum.

5.4 Transcript of the Interview with Participant P1

Interview

Interviewer: Zurab Chkhitunidze
Interviewee: P1

1. How do you develop and share ideas in a group?

   - I often make suggestion and expect feedback
   - research similar areas
   - paper notes
   - sketches

2. *Do you use any particular techniques?*
   (ex. Brainstorming, mind map, Sticky notes, notes apps)

   I write ideas on several paper notes, then sort and store them.

3. *Do you use any software that helps you in this process?*
   (Google Docs/Sheets/Keep, chat groups, Mind Map Apps, Notes Apps)

   Mostly Google Docs, Google Keep and Whatsapp Groups

4. *What do you expect from such software?*

   - I want to learn about the rules in the app
   - I want to be able to chat in the app with others
   - there should be a feature to anonymize names of participants
   - I would also like to be able to use this software alone
   - It must be easy to add new ideas

5. *Would you use such software for your personal and work projects? Why?*

   Yes, It would help me to work on ideas with groups remotely. It would especially be helpful for me now, because of the pandemic.

6. *Further suggestions and recommendations related to the topic?*

   - the app must be simple, easy to use
   - people should be able to use it in groups or alone
   - there must be tutorials or other kind of guidance
   - the rules must be clear

Note: original Interview was conducted in German language. These are key points (summary) of the interview translated into English.

5.5 Transcript of the Interview with Participant P2

Interview

Interviewer: Zurab Chkhitunidze

Interviewee: P2

1. *How do you develop and share ideas (personally, at work)?*

   My method of developing ideas depends on the situation. I like to share my ideas with others directly (verbally).

2. *Do they use any particular techniques?*
   (ex. Brainstorming, mind map, Sticky notes, notes apps)

   I mostly use mind maps, Agile Cafe.

3. *Do you use any software that helps you in this process?*
   (Google Docs/Sheets/Keep, chat groups, Mind Map Apps, Notes Apps)

   Mostly Google Docs, Chat Groups.

4. *What do you expect from such software?*

   - easy to use
   - it should be asynchronous
   - I'd like to have in-app guidance

5. *Would you use such software for your personal and work projects? Why?*

   - It could save me time
   - for group sessions, it would be helpful for me, if the app does the guidance for everyone
   - the app would help me and my team find best solutions
   - would make my work somewhat easier

6. *Further suggestions and recommendations related to the topic?*

   I would like the app to notify me about new important project updates.
   I would like to be able to control the time of the ongoing project.

Note: original Interview was conducted in German language. These are key points (summary) of the interview translated into English.

5.6 Transcript of the Interview with Participant P3

Interview

Interviewer: Zurab Chkhitunidze

Interviewee: P3

1. *How do you develop and share ideas (personally, at work)?*

   I mostly make sketches or index cards (printed or handwritten).

2. *Do they use any particular techniques?*
   (ex. Brainstorming, mind map, Sticky notes, notes apps)

   At work I always make a schema that shows how the project should be built. I often write down Ideas as keywords instead of whole sentences.

3. *Do you use any software that helps you in this process?*
   (Google Docs/Sheets/Keep, chat groups, Mind Map Apps, Notes Apps)

   Many of them: Google Docs, Dropbox, Skype, Video Chats, Google Calendar,

4. *What do you expect from such software?*
   I expect to have a tutorial that explains how to use the software. Preferably in keywords and steps.

5. *Would you use such software for your personal and work projects? Why?*
   Yes, I often use Google Docs for collaboratively working on ideas but I find it not clear and sometimes confusing. I would like to use a more simple and easy to use tool.

6. *Further suggestions and recommendations related to the topic?*
   The software should be:
   - easy to use
   - easy to understand
   - have intuitive interface

Note: original Interview was conducted in German language. These are key points (summary) of the interview translated into English.

5.7 Transcript of the Usability Test with Participant P4

Usability Test

Date: 16.01.2021
Interviewer: Zurab Chkhitunidze
Interviewee: P4

Original Interview was conducted in German language. These are key points (summary) of the interview translated into English.

Think Aloud phase

1. **Invitation code screen**
   I understand the purpose of the elements on this page (invitation code input field and next button).
   [enters invitation code, presses next]

2. **Welcome screen**
   The explanatory text and the purpose of the "go to experiment" button are understandable. I wonder if I can see this text later again.
   [presses "go to experiment" button]

3. **Ideas list screen (with 0 ideas)**
   I would like to be able to see the welcome text again anytime in the app (for example by pressing a "info button" on the top of this screen).
   I will probably see existing ideas on this screen when there are some.
   Over the "+", I will probably be able to create a new idea.
   [presses the "+" button]

4. **Create idea screen**
   The text input field labeled "idea" is most probably the title or short form of a new idea. I see a character limit indicator on the bottom of the input field (70). I think 70 characters should be enough.
   [types new idea in the text field]
   The "add comment" element does not really look like a button. Is it clickable?
   [presses "add comment" button]
   I think this element should look like a button. It looks more like a non-clickable text.
   [presses the "save" button]

5. **Ideas list screen (with 1 idea)**
   I see my idea in the list. I think that it is good that I can see the author of the idea below the title. Number of ratings and number of comments would be also helpful.
   [holds a mobile phone next to the prototype]
   I think that the texts under the idea title are too small.

I would like my ideas to be marked differently and also to have a possibility to sort the ideas by different criteria (most comments, most ratings, most recently added, etc.).
[presses on the idea card]

6. **Idea overview screen**
   If I click the “rate” button on the top I expect to be able to give my rating to this idea.
   [presses “rate” button on the top]
   It is intuitive, but I think that it is not necessary to be able to rate a non-whole number of stars (for example 4.5). The initial value of the rating (3.5) may influence my rating. Maybe 0 Stars as the initial rating would be better.
   [presses “rate” button in the rating popup]
   [presses “rate” button on the top again]
   At this point I expect to see the rating I gave to this Idea instead of the initial one again.
   [presses on “add comment” input field]
   [writes a comment and presses “send” button]
   Adding comments is intuitive.

Additional Feedback Phase

- In the ideas list the text “not rated” should take as much space as the stars when the idea is rated.
- I would like to see the list of the participants
- The participants must be able to edit and delete their ideas.
- Ideas list should have a sorting feature (sort by “rated by me first”, “best rated”, “last activity”)
- There should be some sign that indicates that I already rated the idea
- Number of comments and ratings should be visible on every idea card on the list.
- I would like to have a list of recent activities.
- “Notify Me” feature that sends me notification for new activities on a particular idea.
- After a click on a system notification of this app (while I am not using this app) I expect to be navigated to Ideas list screen.
- The system back button should always navigate the user to the previous page.

5.8 Transcript of the Usability Test with Participant P5

Usability Test

Date: 19.01.2021
Interviewer: Zurab Chkhitunidze
Interviewee: P5

Original Interview was conducted in German language. These are key points (summary) of the interview translated into English.

Think Aloud phase

1. **Invitation code screen**
   Invitation code input field and “next” button are clearly understandable. The button labeled “login as experimenter” might be confusing for some participants.
   [enters invitation code and presses next]

2. **Welcome screen**
   I am on the page with explanatory text for the project. After clicking on the “go to experiment” button I expect to proceed to the next page where I can start generating ideas.
   [presses “go to experiment” button]

3. **Ideas list screen (with 0 ideas)**
   As the title of the page is “ideas”, I think I will see committed ideas on the I understand the purpose of the “+” button, I will be able to add a new idea after clicking on it. Maybe clicking on the empty field in the middle could lead to the same function?
   [clicks in the middle of the empty field]
   [presses the “+” button]

4. **Create idea screen**
   I think the text input field is for the short form of the idea.
   [types new idea in the text field]
   Is “add comment” a button?
   [presses “add comment” button]
   This should look more like a button. Actually the second input field could be always visible. Clicking on a button to make it visible is not necessary. If I click the “save” button, I will go to the previous screen.
   [presses the “save” button]

5. **Ideas list screen (with 1 idea)**
   I see the list of generated ideas with one idea that I recently commited. It is clearly understandable. I would like to see the number of comments here.

[presses on the idea card]

6. **Idea overview screen**
   I would click the “rate” button on the top to rate this idea. I expect to be able to rate an Idea based on some scale (in a particular range) or with one of two options (for example “like” and “dislike”).
   [presses “rate” button on the top]
   The initial value of 3.5 stars is ok but generally it is sufficient to rate with only a whole number of stars.
   [presses on “add comment” input field]
   [writes a comment and presses “send” button]
   Adding comments worked as I expected. Existing comments are also displayed in an understandable way.

Additional Feedback Phase

- Maybe it would be helpful if users also could rate comments (like or dislike).
- I think that color palette is appropriate.
- I would like to have a feature to group or sort the ideas.
- Navigation of the app is understandable.
- It must be possible to delete ideas. This could be accomplished by a long press on an Idea card. In this case the card should be highlighted and a “delete” button should appear on the top.
- When adding ideas it would be better to add a description instead of first comment.