

Master Thesis



Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Computer Graphics and Interaction

Applications for experiments exploring symmetry features in images

Bc. Martin Jandek

Supervisor: Ing. Ivo Malý, Ph.D.
Field of study: Open Informatics
Subfield: Human Computer Interaction
May 2023

Acknowledgements

I would like to express my gratitude to my supervisor, Ing. Ivo Malý, Ph.D., for their guidance and support throughout my Master's thesis project. I am grateful for their expertise, encouragement, but mainly for their patience with me.

I would also like to thank my friends and family for their love and support. Without them, this journey would not have been possible.

Declaration

I, Martin Jandek, hereby declare that this Master's thesis has been solely composed by me, under the guidance of my advisor, Ing. Ivo Malý, Ph.D., and in accordance with the rules and regulations of the Czech Technical University in Prague, Faculty of Electrical Engineering. This work represents my original research, not previously submitted for any other degree, and any external sources utilized have been appropriately acknowledged. I take full responsibility for any errors, omissions or inconsistencies in this thesis and declare that the findings, conclusions, and recommendations presented are solely based on my research. I grant Czech Technical University the right to reproduce and distribute copies of this thesis for archival purposes while retaining the copyright. I have adhered to all ethical guidelines and regulations throughout this research.

Martin Jandek

In Prague, 26. 5. 2023

Abstract

This master's thesis discusses the design and implementation of a web application for administration and completion of psychophysical experiments, primarily the 2-AFC tasks. The application handles everything from the creation and administration of experiments, settings of the testing environment, and completion of the tasks on the participant's side. The main goal was to provide a modern looking and user-friendly application, but still a scientific tool. The implementation itself makes use of currently trending frameworks and approaches and focuses on extensibility and sustainability for future support.

Keywords: quantitative research, quantitative experiments, psychophysics, psychophysical experiments, 2-AFC, interactive experiments, symmetry, visual comparison, react, next.js, user testing

Supervisor: Ing. Ivo Malý, Ph.D.

Abstrakt

Tato diplomová práce diskutuje design a samotný vývoj webové aplikace pro správu a vyplňování psychofyzikálních experimentů, primárně takzvaných 2-AFC testů. Aplikace zahrnuje vše od tvorby a správy experimentů, nastavení testovacího prostředí až po samotný průchod experimentem konkrétním účastníkem experimentu. Hlavním cílem bylo nabídnout moderně vypadající a uživatelsky přívětivou aplikaci, která ale stále splňuje požadavky na validní vědecký nástroj. Implementace využívá aktuálně populární frameworky a vývojové přístupy a je zaměřena na rozšířitelnost a udržitelnost pro budoucí údržbu.

Klíčová slova: kvantitativní výzkum, kvantitativní experimenty, psychofyzika, psychofyzikální experimenty, 2-AFC, interaktivní experimenty, symetrie, vizuální porovnání, react, next.js, uživatelské testování

Překlad názvu: Aplikace pro experimenty zkoumající symetrii v obrázcích

Contents

1 Introduction	1	6 Testing the Application	35
2 Task Analysis	3	6.1 First Round of Usability Testing	35
2.1 Qualitative and Quantitative Research	3	6.1.1 Background Summary	35
2.1.1 Qualitative Research	3	6.1.2 Methodology	35
2.1.2 Quantitative Research	3	6.1.3 Results	36
2.2 Psychophysics	4	6.1.4 First Test Round Feedback	36
2.3 Experiment Definition	4	6.2 Second Round of Usability Testing	37
2.3.1 2-AFC Task	4	6.2.1 Background Summary	37
2.4 Chapter Summary	6	6.2.2 Methodology	37
3 Application Requirements	9	6.2.3 Results	37
3.1 Requirements	9	6.2.4 Second Test Round Feedback	38
3.1.1 Functional Requirements	9	6.3 Third Round of Usability Testing	38
3.1.2 Non-Functional Requirements	10	6.3.1 Background Summary	38
3.2 Existing Solutions	10	6.3.2 Methodology	39
3.2.1 "Versus" tool	11	6.3.3 Results	39
3.2.2 Qualtrics	12	6.3.4 Third Test Round Feedback	39
3.2.3 jsPsych JavaScript Framework	13	6.4 Findings and Changes in the Application	41
3.2.4 PsychoPy	14	6.5 Chapter Summary	41
3.2.5 Other Solutions	14	7 Results	43
3.3 Chapter Summary	15	7.1 Current State of the Final Application	43
4 Design	17	7.1.1 Researchers' application	43
4.1 Design Approach	17	7.1.2 Participants' application	45
4.2 Design Proposal Iterations	18	7.2 Meeting the Requirements	45
4.2.1 Design Iteration 1 - Sketch	18	7.3 Further Extension	46
4.2.2 Design Iteration 2 - Sketch	20	7.4 Chapter Summary	48
4.2.3 Design Iteration 3 - Sketch	21	8 Conclusion	49
4.2.4 Design Iteration 4 - HTML	22	A Application Manual	51
4.3 Chapter Summary	23	A.1 Create and Configure an Experiment	51
5 Implementation	27	A.2 Setup the Content of the Experiment	51
5.1 Technologies Used in the Application	27	A.3 Finish Experiment Creation and Invite Participants	52
5.1.1 React	27	B Master Thesis Specification	53
5.1.2 Next.js	29	C Bibliography	57
5.1.3 TypeScript	29		
5.1.4 MongoDB	30		
5.1.5 Other Technologies Used in the Application	31		
5.2 Project Structure	32		
5.3 Best Practices	32		
5.3.1 Stateful versus Stateless Components	33		
5.3.2 React Hooks	33		
5.4 Code Quality	33		
5.5 Chapter Summary	34		

Figures

Tables

2.1 Sample of unsymmetrical lenses . . .	5
2.2 Sample of symmetrical lenses	6
2.3 Example of 2-AFC comparison . . .	7
3.1 Main graphical user interface of the "Versus" tool	11
3.2 Example of a Qualtrics creator UX	13
3.3 Example of a Qualtrics report . .	13
3.4 Example of a graphical PsychoPy builder	15
4.1 Design iteration 1 of experiment detail page	18
4.2 Design iteration 1 of section detail	19
4.3 Design iteration 1 of experiment list	20
4.4 Design iteration 2 of experiment detail	21
4.5 Design iteration 2 of section detail	22
4.6 Design iteration 2 of experiment list	23
4.7 Design iteration 3 of experiment list	24
4.8 Design iteration 3 of experiment detail	24
4.9 Experiment settings in the design iteration 3	25
4.10 Design iteration 3 of section detail	25
4.11 Design iteration 4 of experiment list	26
6.1 The list of experiments divided into locked and unlocked subcategory . .	38
7.1 Login page	44
7.2 Main page with list of created experiments	44
7.3 Detail of an experiment	45
7.4 Detail of a blank section	46
7.5 Detail of experiment results table	47
7.6 Participant's view of a 2-AFC comparison	47
7.7 Page with the option to withdraw participant's consent	48



Chapter 1

Introduction

There are few to none existing simple to use free tools, which do not require coding skills that could be used for academic visual psychophysical experiments. Recently, the popularity of moving from laboratory user testing to an online and more participant-friendly space has grown, and there is a sensible need to have access to a set of tools for that purpose [10]. Human symmetry detection and more specifically estimating a threshold, where the human mind detects a symmetry, by quantitative research, is, for example, one of the experiments that could be conducted even remotely. This thesis deals with that problem and openly collaborates with a team consisting of members from DCGI, Czech Technical University in Prague, and FAS, University of West Bohemia, examining human symmetry detection.

A study suggests that JavaScript is a viable tool for conducting time-precise psychophysical experiments [10, 13]. There are a few possibly suitable tools for conducting such experiments (see section 3.2), but none of them meets all the requirements for a simple, user-friendly but still scientific tool. This paper sets out to address the gap and strives to propose and develop a solution. I have analyzed the existing tools, proposed and designed my own solution, and developed and user tested a functional tool for running quantitative psychophysical visual experiments.

First, in chapters 2 and 3, I define the task and analyze possible approaches and state-of-the-art solutions. In chapter 4, I outline a specific design and describe its implementation in chapter 5. Finally, on a few possible end-users, we test the delivered design and convey the feedback provided in chapter 6.

Chapter 2

Task Analysis

2.1 Qualitative and Quantitative Research

Before diving into the waters of the one particular research, on which this thesis is based, I will provide a brief description of two main research approaches, their differences and goals [19, 35, 7].

When collecting and analyzing data, research in general can be divided into qualitative and quantitative. Both are very important, and often, both are performed during a particular study. Each type of research has its qualities and goals, which are very different.

2.1.1 Qualitative Research

Qualitative research aims to explore and understand some facts using detailed examination and interpretation of non-numerical data, often in the form of natural language. It employs a holistic approach and focuses on subjective experiences and feelings and their meaning in both the context of the research and the social context.

Qualitative research makes use of techniques such as interviews and observations and produces rich and descriptive data as a result. It allows researchers to better understand the details and complexities of human experiences during the study. The main focus of this technique is to explore the study and other ideas in more depth and to formulate hypotheses. Due to the complexity of the qualitative research process, only a few respondents are needed to carry out a valid qualitative research.

I used a qualitative research approach during this thesis for the purpose of usability testing of the final application, the result of this work. Thus, an example of the qualitative research can be found in chapter 6.

2.1.2 Quantitative Research

Quantitative research focuses on the systematic collection and analysis of numerical data. The methodology of quantitative research should provide objective and replicable results, without subjective meaning.

Quantitative research employs techniques such as surveys and experiments, focusing on measurable variables and producing data in numerical forms. On the basis of the data, researchers utilize statistical tools for the analysis. They identify trends, establish cause-effect relationships, and make predictions. The main focus of this research method is to generate replicable numerical data or to test hypotheses. To make a valid use of statistical methods, a large number of respondents and big data are needed to perform such research. Quantitative research often takes place after qualitative research. Using qualitative research first, we can form some ideas and formulate some hypotheses which we then test using quantitative research.

2.2 Psychophysics

Psychophysics is a branch of psychology that studies the relationship between physical stimuli and the perceptions they produce. Psychophysical experiments are designed to explore this relationship by changing the stimuli and observing how the changes affect a person's perception of the stimuli. One branch of psychophysics is visual psychophysics. Visual psychophysics uses only visual stimuli and investigates human perception.

2.3 Experiment Definition

In this thesis, we study visual stimuli. More specifically, we study generated symmetry pictures. We use software [14] that generates images with randomly placed lentils (dots, points). These dots are either perfectly reflection-symmetric or with an imperfection manually defined by percentage. By varying the level of imperfection, we can investigate the decision point at which a person perceives symmetry. In fig. 2.1 we can see an example of imperfect symmetry. The symmetry axis is vertical in the center of the image and there are added lentils on the left side of the figure. In comparison, we can see a perfect symmetry in fig. 2.2. The dots in this image are perfectly symmetric along the central vertical axis.

2.3.1 2-AFC Task

One of the most widely used psychophysical tasks in visual psychophysics is the Two-Alternative Forced Choice (2-AFC) task [4]. It requires the participant to choose one of two alternative stimuli depending on the research question. We can see a sample of such comparison in fig. 2.3.

In this way, the researcher presents the participant with a series of questions containing images with different levels of symmetry. As a result of the 2-AFC task, we can either create a ranking of the images or approach a threshold estimation [45, 40].

It is important to note that the handling of the 2-AFC task results is beyond the scope of this work and is entirely up to the researcher conducting

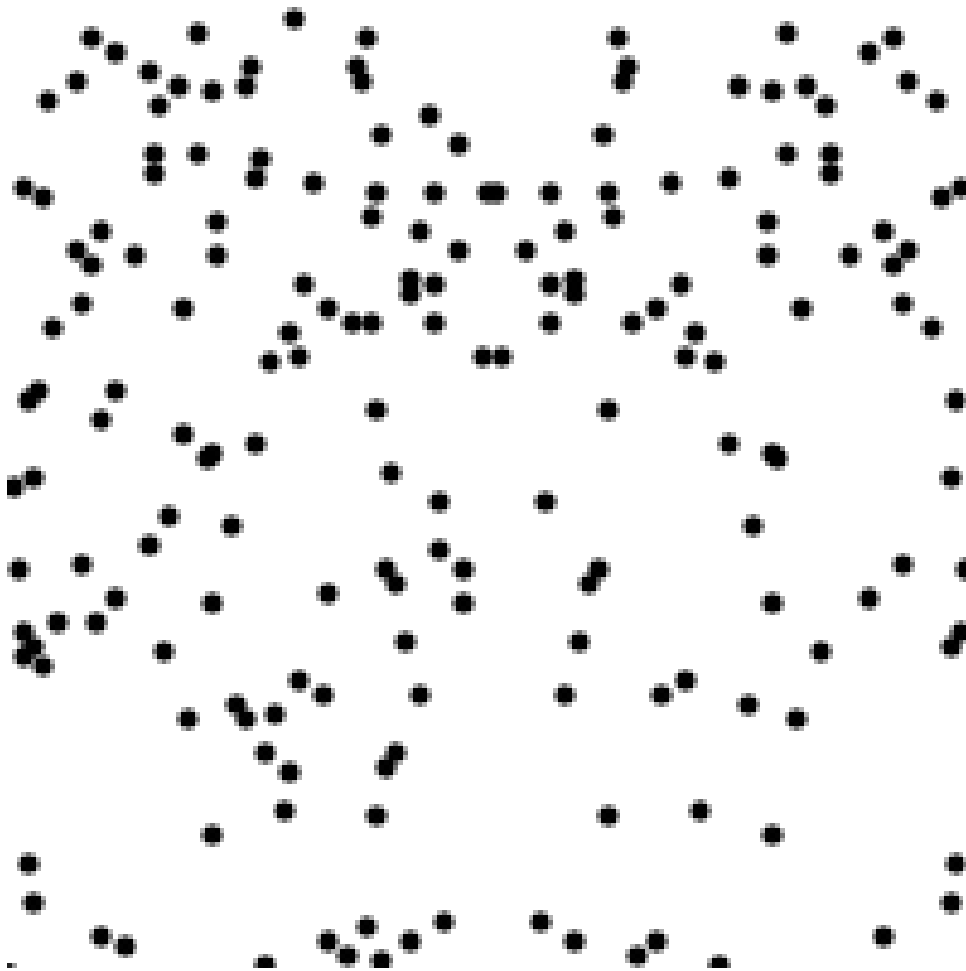


Figure 2.1: Sample of generated lentils with imperfect symmetry (symmetrical along the central vertical axis with more lentils on the left side)

the experiment. In some experiments, it may be important to create a ranking system for the images tested. In other situations, it may be more beneficial for the researcher to know only the results of each individual question asked during the 2-AFC task.

■ Participant Environment

It is important to consider the participant and the experimental environment. Since a human can detect symmetry efficiently and quickly, the experimental environment should be controllable. The researcher needs to control the position and size of the images and the distance between them. The participant must have the tested images in his or her primary field of view. Also, symmetry detection is a very fast process, and the researcher should be able to set a time limit for the comparison questions.

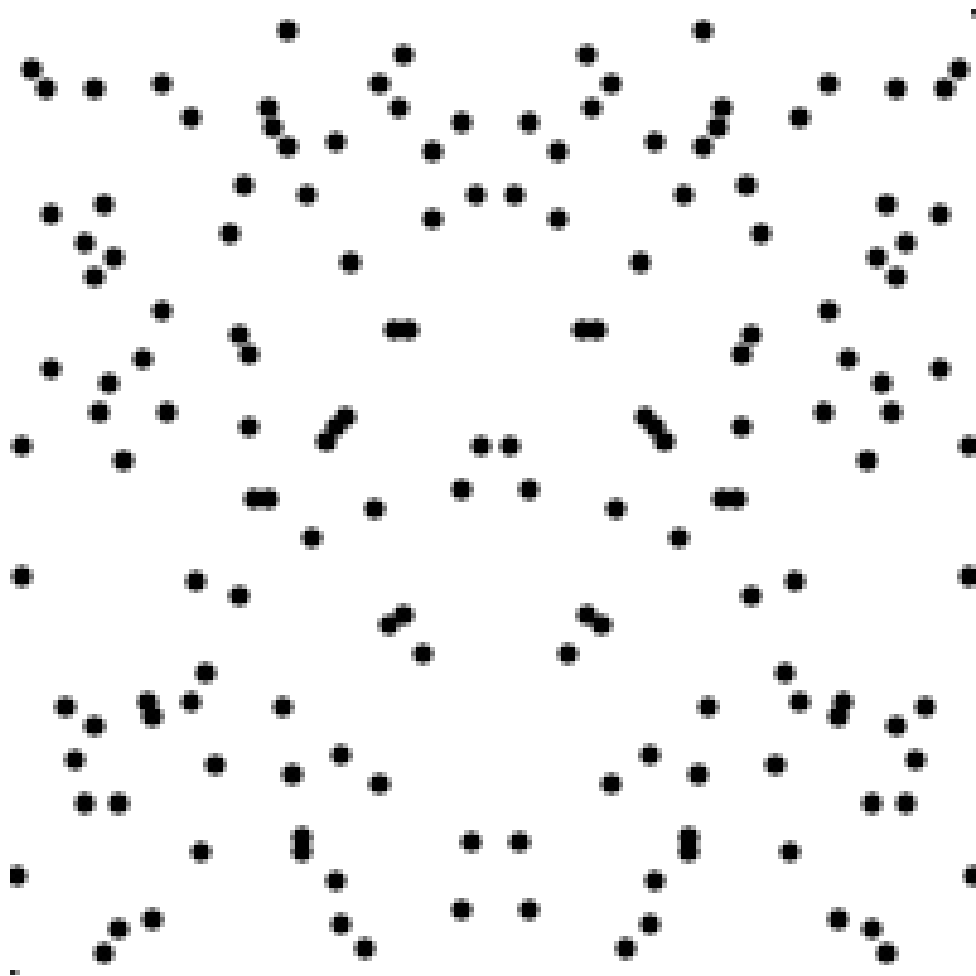


Figure 2.2: Sample of generated lentils perfectly symmetrical along the center vertical axis

■ 2.4 Chapter Summary

In chapter 2, I have analyzed the motivation behind the thesis topic and described the visual psychophysics task mainly used, the 2-AFC task. From the analysis I can now in section 3.1 draw the first few requirements of the to-be-designed tool for creating the experiments.

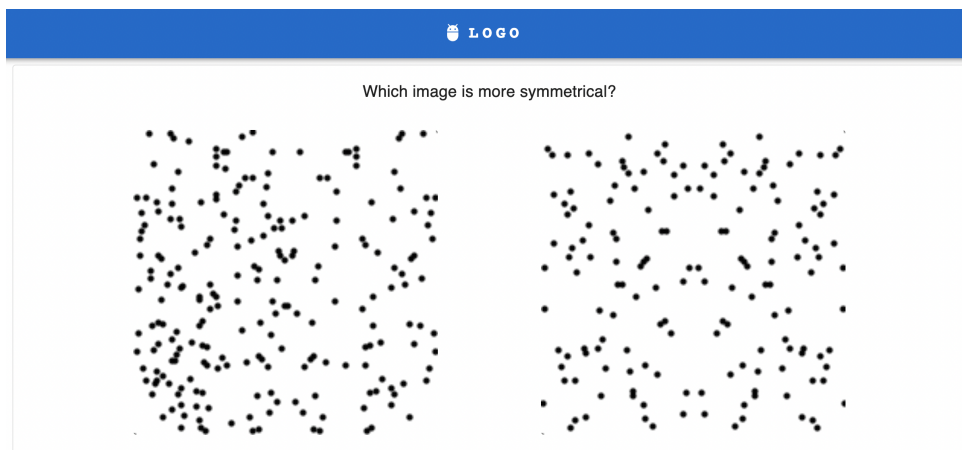


Figure 2.3: Example of comparison of two visual stimuli during 2-AFC experiment

Chapter 3

Application Requirements

3.1 Requirements

Before I start designing the application, I need to specify the requirements and possibly add some non-functional requirements to guide the feel of the final design.

3.1.1 Functional Requirements

First, I will describe the functional requirements that result directly from the assignment of the thesis and the description of the task. I will then list some additional requirements that emerged from a discussion.

Requirements Resulting From the Thesis Assignment

1. The tool allows the researcher to create a 2-AFC user testing experiment.
2. The tool allows the researcher to set up the types of question in the experiment and their order.
3. The tool allows the researcher to add an interactive question (such as drawing the axis of the symmetry to the image by the participant) to the experiment.
4. The tool allows the researcher to access and evaluate the results of such experiments.
5. The tool allows the researcher to export both the result of each individual question raised during the experiment and the result of the whole experiment.
6. The tool allows the researcher to administer experiments and participants.
7. The tool allows participants to log in and fill in the experiment intended for them only (login of a particular participant or some alternative).
8. The tool allows multiple user access for researchers.

9. The researcher's and the participant's parts of the tool are strictly divided.

■ Additional Requirements

1. The experiment is divided into sections, each section is divided into questions. Each section can examine different stimuli and can use different research question.
2. The tool enables the researcher to export the experiment results to CSV.
3. The tool sends an email invitation to the participant on demand of the researcher.
4. The tool allows the participant to reject the usage of his results and to delete the obtained data related to him.

■ 3.1.2 Non-Functional Requirements

I will also list some other requirements that are not directly part of the assignment, nor are they part of the functionality, but the final application should meet them.

1. Web application
 - The final designed tool, for reasons of high availability, should be a web application working in the latest most-used browsers.
2. "Big display" priority
 - The researcher's part of the application should be optimized mainly for computer use, but it should not be unusable on mobile devices.
3. Anonymity of the participants
 - No experiments or results should be linked to a specific participant.
4. Security in mind
 - Along with the previous point, keep in mind that the researchers work with sensitive data.
5. Ease of use
 - The UX of the application should be designed with the user in mind and along with the state-of-the-art UI designs.

■ 3.2 Existing Solutions

Before designing the application according to the analysis and the requirements, I will first analyze the possible existing applications and frameworks and summarize its advantages and disadvantages.

3.2.1 "Versus" tool

The tool called Versus [42] is probably the one that would tick the most checkmarks on the requirements list. Versus is a tool developed to compare visualization techniques using the 2-AFC task. It is a web-based application that can run locally or remotely.

When comparing Versus to our requirements and needs, it allows a researcher to create 2-AFC experiments. However, it is not possible to control the settings of the experiment, the individual sections, and the questions of the experiment or the group of participants. The tool allows one to upload a bulk of images as "imagesets" and it generates all comparisons without further settings. The researcher is only able to set an introductory and ending titles and a question header that is visible for each comparison. One of the nice-to-have features to consider for our application is the direct integration of Amazon's Mechanical Turk.

The tool was mainly developed to create a ranking system of visualization techniques. It uses an optimization technique when generating comparisons using a sorted binary tree structure and assuming transitivity (when $A > B$ and $B > C$, then we do not need to compare A with C and assume that $A > C$) [30]. Because we do not create a ranking system from the results and we need the full 2-AFC task results, the optimization technique is not something to consider for our use case.

Parameters

Fraction to be Repeated
0.2

start header
Welcome to Versus!

start blurb
You will now start the survey. Do not forget to copy the survey code provided at the end of the survey, and paste it into the Mechanical Turk page.

running header
Which image do you prefer?

running blurb
Please select the image you like best.

done header
Thank you for participation.

done blurb
Your tests are done. Thank you. Your survey code is

UPDATE TEXT

Participants

Mechanical Turk Survey Link: [/#/mechanical-turk/5](#)

INVITE PARTICIPANT DOWNLOAD "RESULTS.CSV"

Invite	Status	Code	Time (s)	Answers	Fraction of Repeats	Repeats	Consistent	Created	Updated
link copy	done	B1qPvTsAG	33.1	37	0.2	7	2	Fri May 18 2018 13:23:27 GMT+1000 (AEST)	Tue Jun 05 2018 14:42:57 GMT+1000 (AEST)

Images

1b3u

1: 1b3u_7.png 2: 1b3u_16.png 3: 1b3u_22.png 4: 1b3u_44.png 5: 1b3u_52.png 6: 1b3u_59.png 7: 1b3u_62.png 8: 1b3u_74.png

Figure 3.1: Main graphical user interface of the "Versus" tool

Advantages.

- Web-based application that can be run either locally or remotely
- Very straightforward and simple usage
- Allows a fraction of comparisons to be repeated
- Direct integration of Amazon's Mechanical Turk
- Optimization technique reducing the number of comparisons needed to create an image ranking

Disadvantages.

- No setting options
- Focuses on creation of ranking system, thus not comparing all pairs of images

■ **3.2.2 Qualtrics**

Qualtrics [29] is a powerful web-based tool to create and distribute online user surveys. It uses a very simple point-and-click interface and tens of question types to use, such as a form field, multiple choice field, or slider. Qualtrics provides multiple templates from which a researcher can choose and build the experiment instead of creating one from scratch. Additionally, Qualtrics recently added an additional tool to create psychophysical surveys, or it can also be combined with other psychophysical frameworks. I will not cover the full psychophysical survey examination here because most of the complex functionality is hidden behind a paywall.

Unfortunately, the Qualtrics survey builder is a closed-source project and a paid account is needed to use the full potential of it. The Qualtrics application is a tool that can handle all surveys from employee and product satisfaction to psychophysics experiments including the result analysis and reporting. With that power comes a great cost of a tool "too powerful" and complex for our use-case and expensive to be used for research purposes.

However, the overall UX of Qualtrics is very nice and functional. Handling of the survey "blocks" and block "questions" including the individual settings is very intuitive and clear. The blocks and questions logic from Qualtrics served as an implicit design inspiration for our project design.

Advantages.

- Cloud-based enterprise application that needs no installation or setup
- Handles all sorts of surveys from employee and product satisfaction to psychophysics experiments
- Contains tens of question types and templates
- Allows to create complex surveys with loops and different display logics
- Contains data analysis and visualization tools

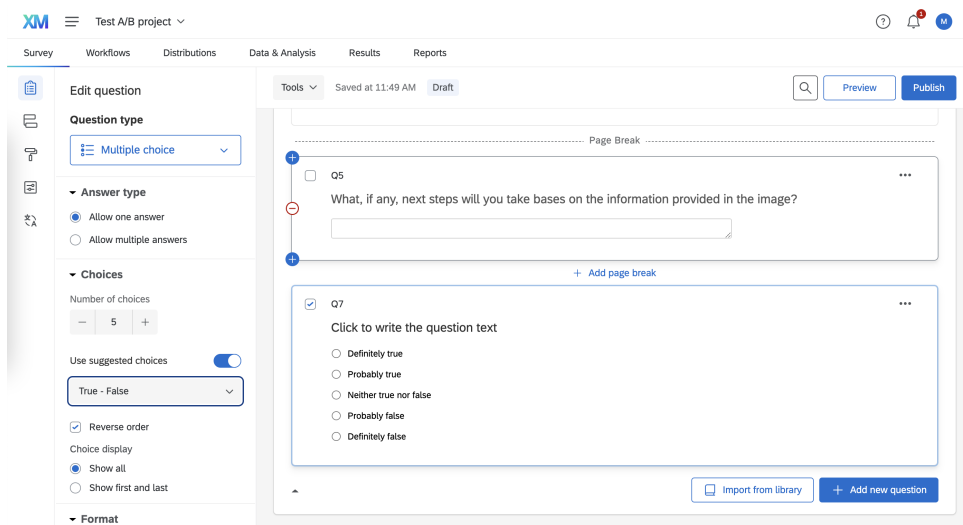


Figure 3.2: Example of a Qualtrics creator UX

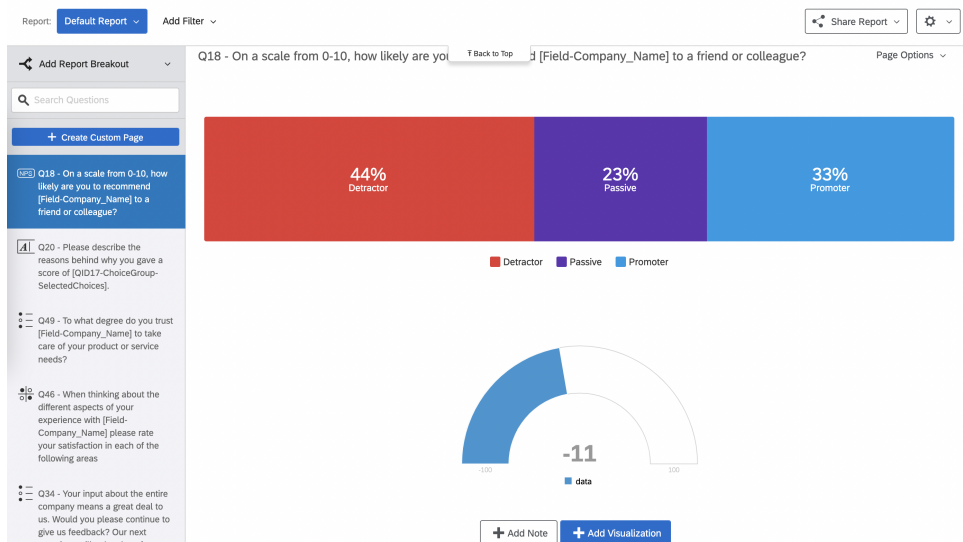


Figure 3.3: Example of a Qualtrics report

Disadvantages.

- Expensive
- Too complex for simple 2-AFC experiments
- Unknown psychophysics capabilities - mostly use for enterprise use-cases

■ 3.2.3 jsPsych JavaScript Framework

With jsPsych [9, 8], it is a dive into the waters of libraries and frameworks instead of full applications.

JsPsych [9, 8] is an open source JavaScript framework, whose purpose is to accelerate the development of a full web-based application for psychophysical experiments. It handles the logic of displaying the stimuli, measuring the response time, and capturing the participant choice. It builds on the expandability and modularity, and if a researcher needs something that core jsPsych cannot do, he can write a plugin for it. It is important to note that jsPsych does not handle data storage at all. When using this framework, one still needs a database or some storage to store the resulting data.

Advantages.

- Simple to use, functional API
- Modular and easily expandable using plugins
- Handles general experiment functionality like displaying the stimuli and capturing the participant answer

Disadvantages.

- To be able to create experiments using the framework, a researcher still needs to know how to code and develop an application utilizing the framework.

■ 3.2.4 PsychoPy

PsychoPy [24, 28, 27] is an open source Python software project that provides tools to create and conduct psychophysical experiments. It started as a library and grew into a graphical application, even with a graphical survey builder. The library also provides a way to generate visual and auditory stimuli in real time [26], for example, based on the previous participant's answers.

Advantages.

- Powerful stimuli generation
- Shallow learning curve [25]
- Allows to create multiple different and complex psychological experiments

Disadvantages.

- A researcher needs to be able to code

■ 3.2.5 Other Solutions

In addition, there are other frameworks or applications that are less important for our purposes [10]. I will not describe in detail all of them here, but instead I will provide a reference to some of the most interesting ones. They provide partly the functionality we aim for, but most of them are just frameworks intended to speed up the development of an application with full functionality.

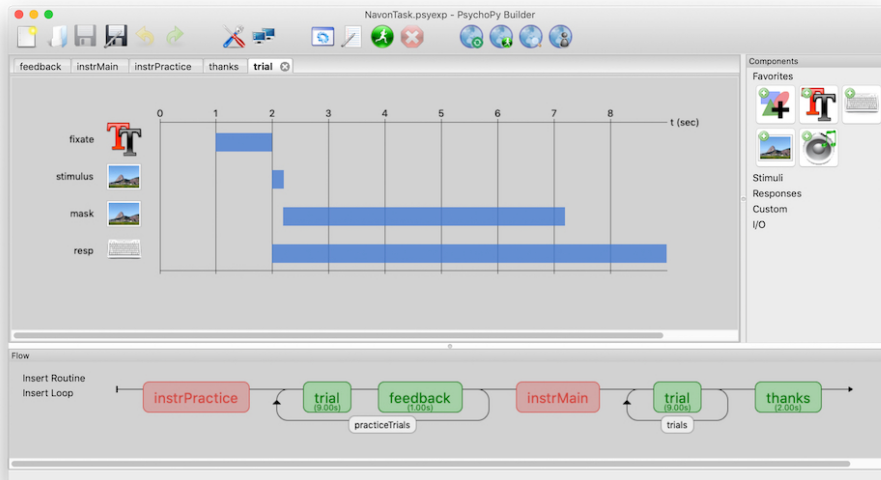


Figure 3.4: Example of a graphical PsychoPy builder

None of them provides the full solution for us to have an administration of multiple experiments at one place.

I decided not to use any of the frameworks to develop the final application for this project. Most of them are open-source, so the final codebase could be controlled, but the functionality they provide would have to be stuck together and integrated into more modern frameworks for web development. Some of them are also intended to be used within a specific environment, like MATLAB.

- PsyToolkit [32, 33, 34]
- MATLAB's Psychophysics Toolbox [15, 3]
- nodeGame [2]
- A platform for subjective image quality evaluation on mobile devices [17]

■ 3.3 Chapter Summary

During analysis of the application requirements that I should meet, I described some of the most important state-of-the-art solutions I used as inspiration. Some of the existing solutions meet some part of our requirements. I took into account the good things from each of the existing solutions described above and considered them during the design phase of this project.

Chapter 4

Design

4.1 Design Approach

Ever since the thesis assignment, I was determined to design a "pretty scientific application". Something that not only works well and is scientifically useful and valid but also looks nice and is easy to use. I decided to take a user-centered design approach and start the design process with possible similarities to already known applications.

Material Design [11] is one of the most widely used and tested approaches to the design of mobile and web applications today. People see it every day in their lives and are familiar with it. It was an easy decision to choose the look of the Material Design applications.

Unlike existing solution designs, I looked for an application that is widely used by many different groups of people and started the design process with something similar. The choice fell on Microsoft's PowerPoint application (as well as any PDF reader, the layout of the main design elements was the most important part of the choice). Both from the requirements and from the discussion with my supervisor I had the idea to edit the experiment and the sections of the experiment as if they were just slides in a presentation. In the end, the completion of an experiment looks like an interactive presentation to the participant.

Based on the specified requirements and the existing solutions designs, I decided to specify three levels of hierarchy. First, the main page for the researcher's overview would be some kind of experiment list. A page displaying brief information about previously created experiments as well as some actions for each of them. The second level would be the experiment details page. There would be an action to go from the experiment list to the experiment detail page. On the experiment detail page there would be settings for the experiment as a whole. As specified in item 1, the experiment would be divided into sections and sections would be divided into questions. Thus, the third level of hierarchy would be the section detail page. Questions and specific section settings would be handled on this page.

We will now look at specific design iterations of these main pages that took place in the design process.

4.2 Design Proposal Iterations

4.2.1 Design Iteration 1 - Sketch

As I suggested, all design proposals are based on Material Design [11]. In the first iteration, I stuck with the top menu bar and decided to go all in with the card layout [12]. As we can see in figs. 4.1 and 4.2, the page would be divided horizontally into three parts, each part wrapped in its own Material Design card (as for the experiment list page in fig. 4.3, each experiment would have its own card). This layout would be the same for both the experiment list page and the experiment detail page (experiment list containing only the experiment cards), the two most important pages in the application. When a user wants to edit the section of a particular experiment (fig. 4.2), he would see a "card within a card" layout, evoking the multi-level hierarchy. I tried to ensure and emphasize the three levels of hierarchy - list of experiments, experiment details, and section details. If the user finds himself in one of the levels, he would know at a glance which level he is in.

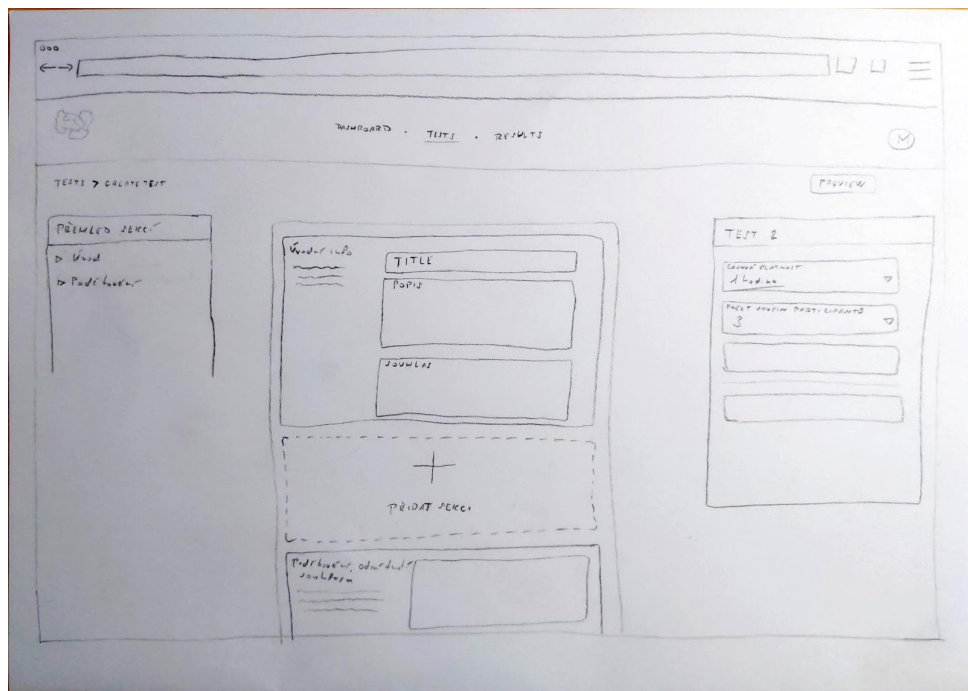


Figure 4.1: Design iteration 1 of experiment detail page. On the left side of the page, there is a list of sections in this experiment. The main, middle part of the page belongs to the sections. We can see the content of each section, we can navigate into the section detail to edit the section content and we can add a new section. On the right part of the page, there are the experiment options that are directly editable.

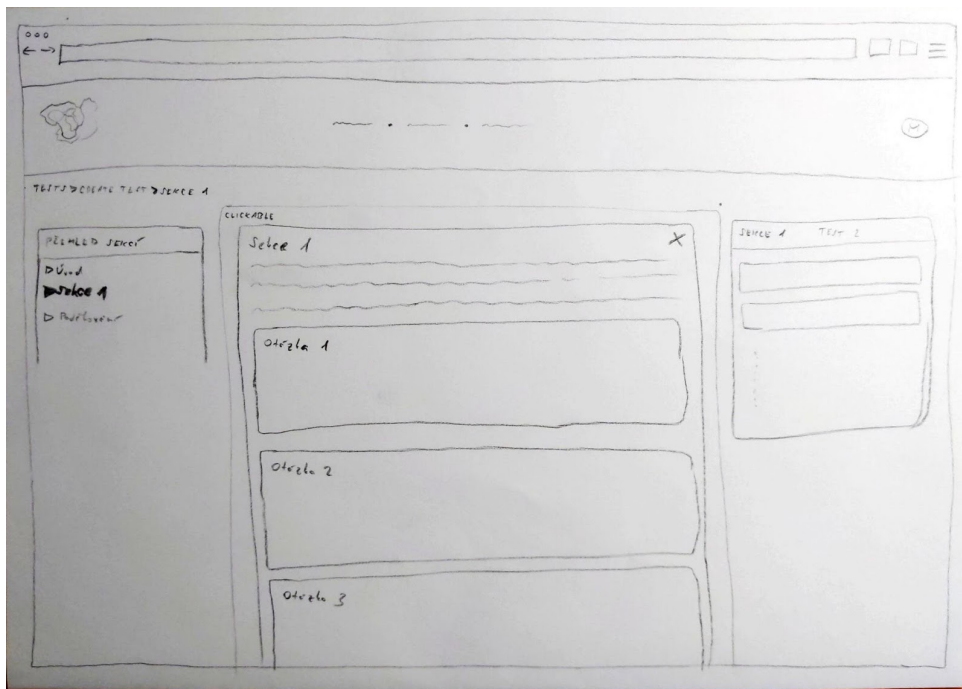


Figure 4.2: Design iteration 1 of section detail. Left part of the page remains the same as for experiment detail - there is a list of sections with the open one highlighted. The main content is wrapped in a "card in a card" layout that displays the hierarchy immersion and the content displays questions in this section. The right part of the page transformed from experiment options to settings of this specific section.

■ UX Expert Review for the First Design Proposal

Having the first design iteration ready, I asked my former colleague, an experienced front-end developer and designer, for an expert UX review. First, I explained the application and gave him some insight into future work with the application. I described the potential researchers who will work with the application and expressed the ideas behind the design choices.

With only three application pages sketched on paper, first, I created some very simple tasks / questions for him to answer.

UX Expert Review Tasks.

1. Looking at this sketch, tell me where you are in the application and what you can do on this page.
2. How would you go from the experiment list to the experiment details?
3. Where would you look if you wanted to change the number of participant groups in the final experiment?
4. Where would you look if you wanted to add a new section?
5. Where would you look if you wanted to add a new question?

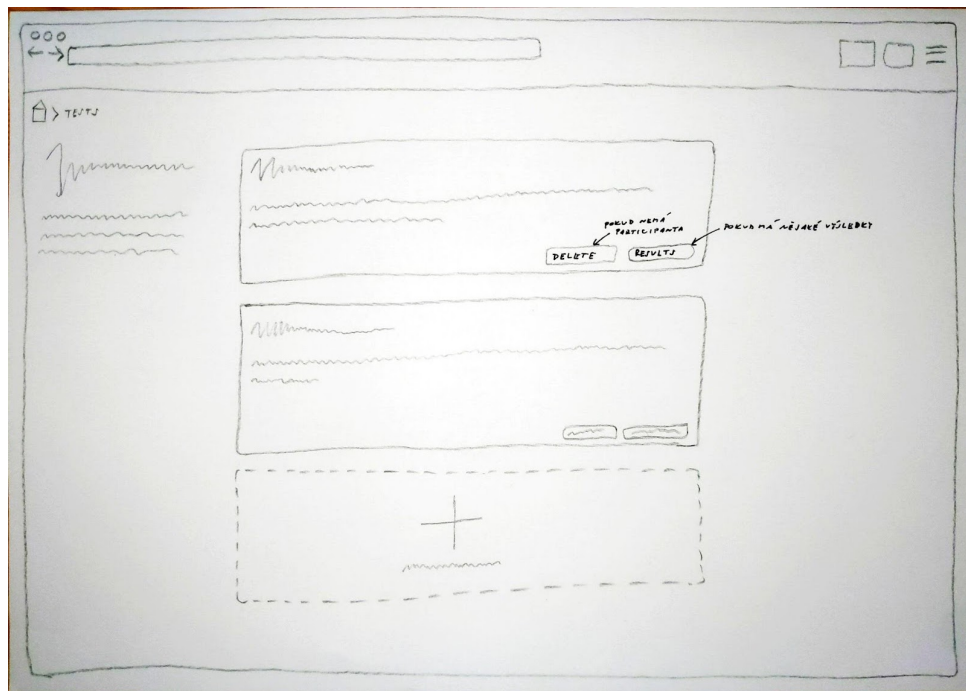


Figure 4.3: Design iteration 1 of experiment list. Left part of the page is dedicated to a page description. In the main center view we can see the list of experiments, each experiment wrapped in its own Material Design card with actions to delete the experiment and to view its results. Clicking the experiment card takes us onto the experiment detail page.

The results of these tasks gave me the first brief feedback before diving deeper into his comments and the subsequent design adjustments. Although the UX expert completed the tasks without problems, there were some problematic parts of the design resulting from the review.

The first and most important change was to remove the editable experiment detail. Having both pages editable was overwhelming and might confuse the user. I decided to keep the experiment detail page only as a preview of each section.

Another change was to remove the "card within a card" layout because it could confuse the user with too much information. The aim was to make the two pages more distinct.

The last change was related to the settings part of the page. Changing the context of the Settings card was again confusing, so I decided to move the global experiment settings to an openable modal instead of having it always visible.

■ 4.2.2 Design Iteration 2 - Sketch

As I said in the previous section, I decided to make some changes in the next iteration. Some of them came from the expert review, others simply made sense to me.

I kept the list of sections as blocks, but made the list uneditable, thus reducing the number of editable levels and making the structure easier to navigate (fig. 4.4). I have moved the experiment settings to a separate modal window, removing the changing right part of the screen. The layers are more distinct, and the settings are more straightforward and separated for the whole experiment and the section (fig. 4.5).

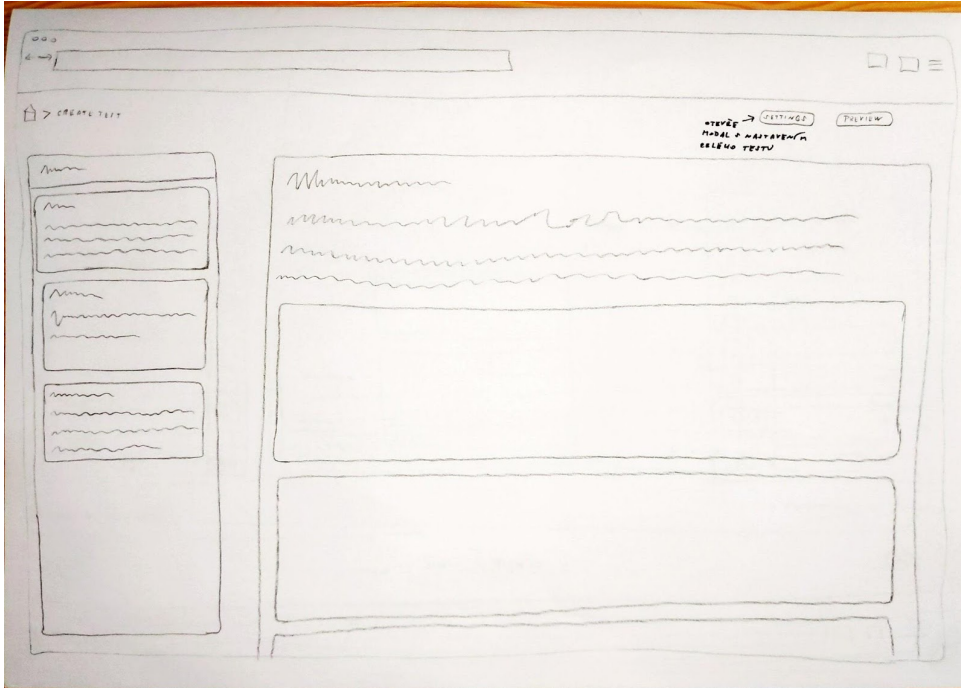


Figure 4.4: Design iteration 2 of experiment detail. The section list on the left side of the page transformed into a card collection view, each section wrapped in its card with title and description of the section. The main part stays the same with some design changes. It shows the title and description of the experiment and a collection of section cards. The experiment settings has been moved to a separate modal that can be open by clicking the "Settings" button located in the top right corner of the page.

For the page of the experiment list in fig. 4.6, I have decided to divide the experiments into categories. After a discussion with my supervisor, it was clear that if the application was to be used as an experiment management tool, there needed to be a way to "deactivate" an experiment and remove it from other active experiments. The design of the experiment list page for both iterations was inspired by the component examples designed by Tailwind UI [36].

4.2.3 Design Iteration 3 - Sketch

In this design round, I decided to change the experiment list page even more (as seen in fig. 4.7). Splitting the experiments into categories in the second iteration was a good idea, but it resulted in the category list being in the

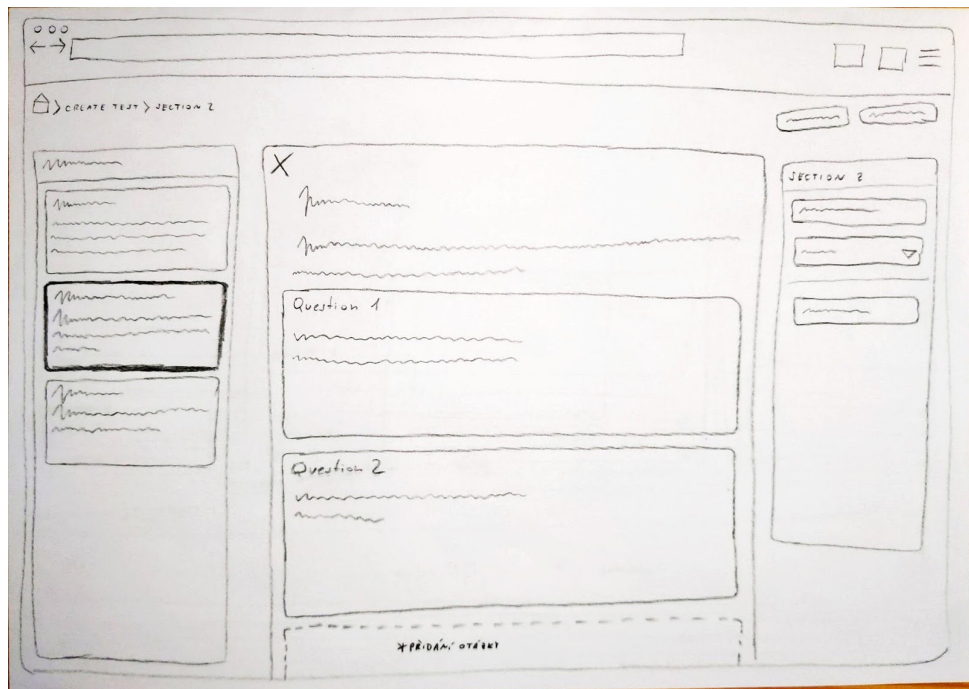


Figure 4.5: Design iteration 2 of section detail. The section list on the left part of the page is the same as on the experiment detail page. Main content is a single card that is now not wrapped in another one. Right part, the section options, stays the same as in the previous iteration.

user's main field of view.

Inspired by the Material guidelines, I sketched the category list card as a "two-line list" from the Material design library. Along with this, I removed the page description on all pages. The page descriptions were patterns used in the Tailwind UI pre-built examples [37], but they did not work well with my design choices and were redundant.

4.2.4 Design Iteration 4 - HTML

At this point, I have decided to start the implementation process and further iterations to build directly using HTML instead of sketches on paper. I generally took a lot of inspiration from the Tailwind UI components and the book "Refactoring UI" [43] by Adam Wathen, the creator of Tailwind.

We can see the HTML implementation of the experiment detail with separate experiment settings in the modal window in figs. 4.8 and 4.9. In fig. 4.10 and fig. 4.11 we can see the section detail and the list of experiments, respectively.

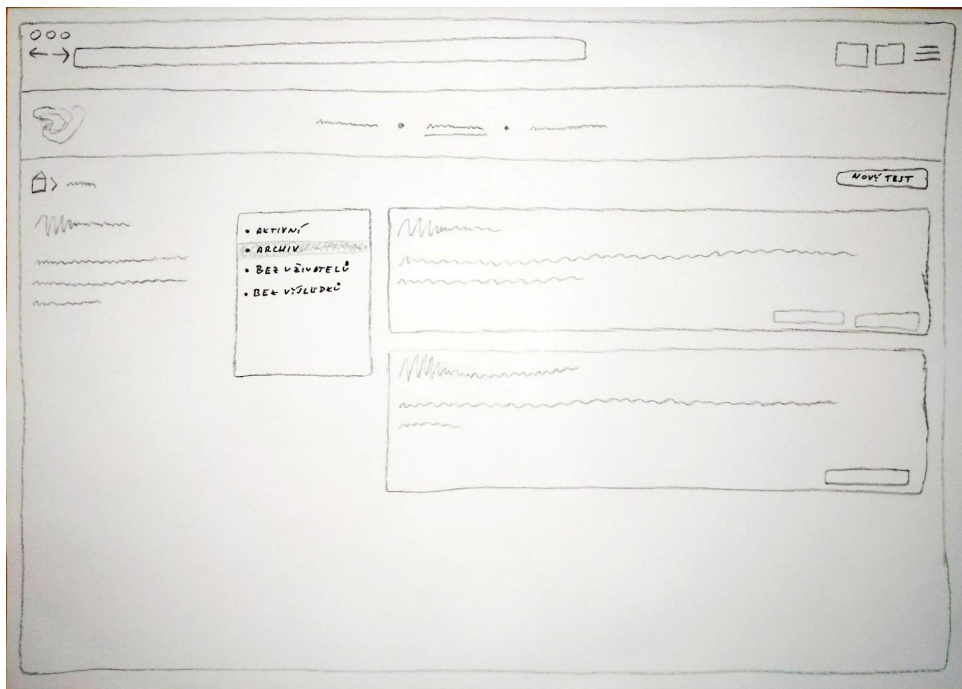


Figure 4.6: Design iteration 2 of experiment list. List of experiment categories (active, archive, etc.) has been added between the page description and the main content, the experiment cards.

4.3 Chapter Summary

I have presented some design proposition and iteratively improved it. Starting with sketches on paper and a broad design idea, and ending with an HTML pages implementing the design choices.

At this point, I focused on the logic implementation, not on the pure design. Minor design changes have taken place since the final design iteration, but no significant changes, so I would not consider it another iteration. We can see the final design in section 7.1, specifically in figs. 7.1 to 7.4.

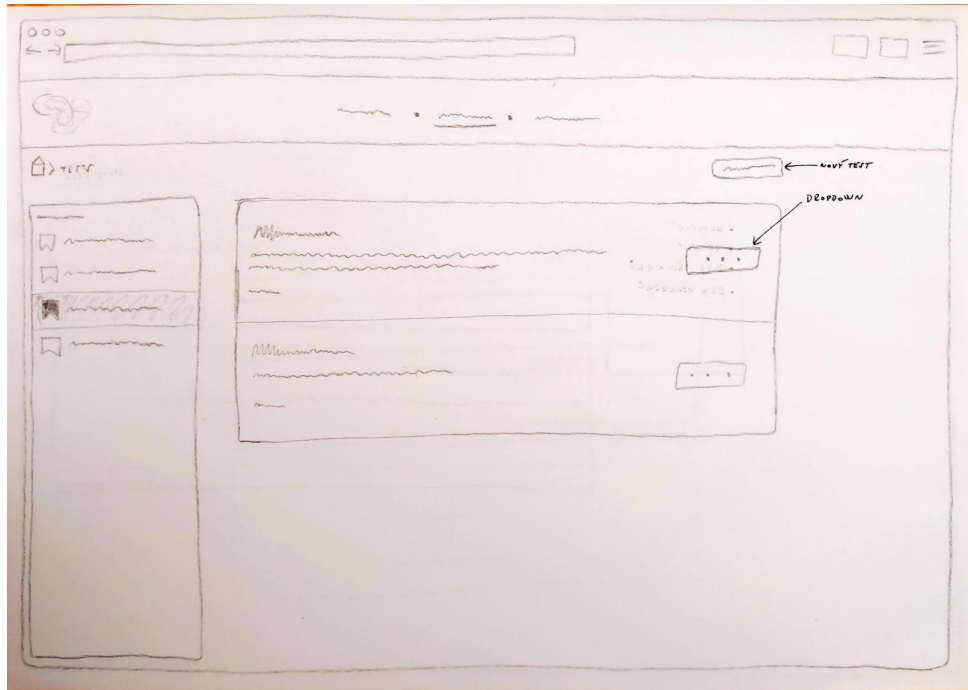


Figure 4.7: Design iteration 3 of experiment list. No page description is needed for this page, thus the list of experiment categories takes place in the left column of the page. Main content is now wrapped in a single card with separate experiments being only list items instead of each experiment having its own card.

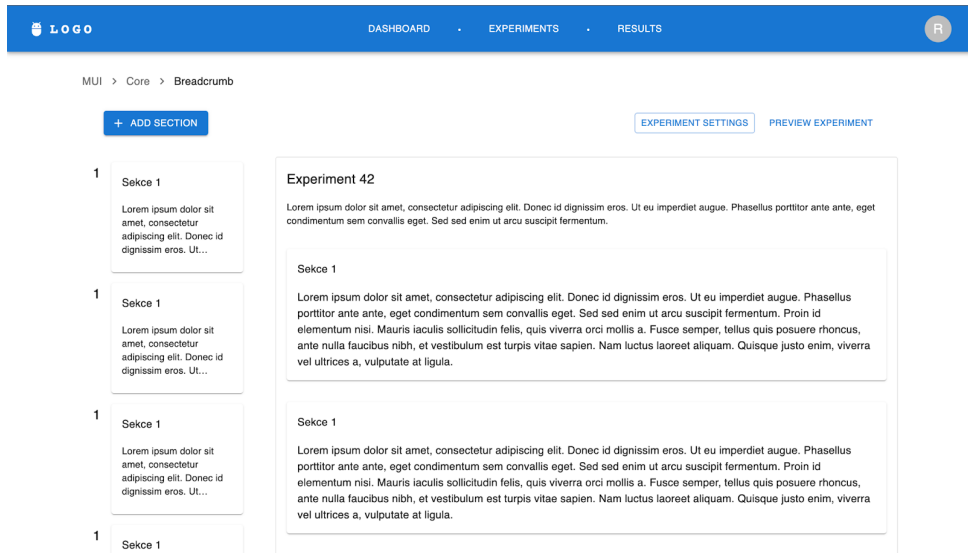


Figure 4.8: Design iteration 3 of experiment detail. The overall design stays the same as in the previous iteration, now designed using HTML with the use of MUI tools instead of just sketches on paper.

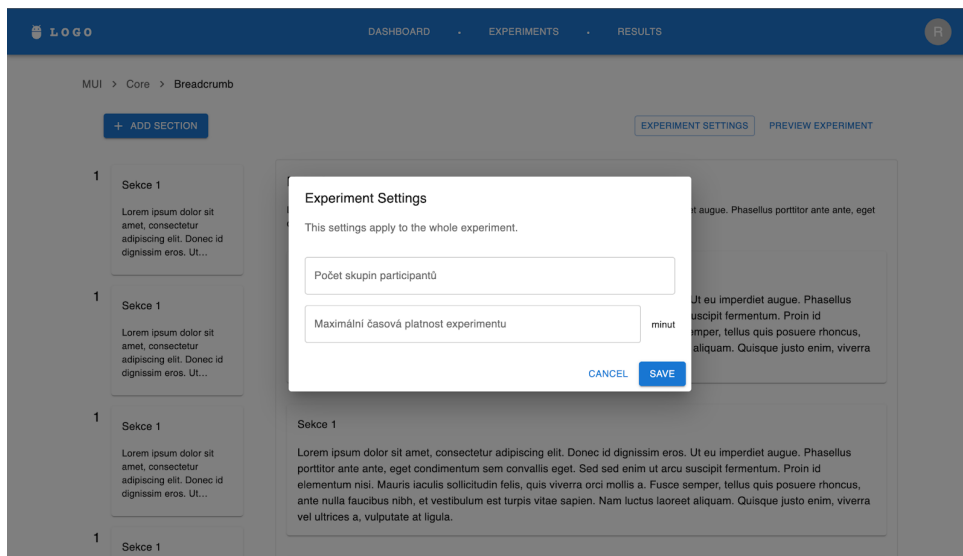


Figure 4.9: Experiment settings in the design iteration 3. The experiment settings that could be seen in the first design iteration has been moved to a separate modal. Here we can see the design proposal of the settings modal.

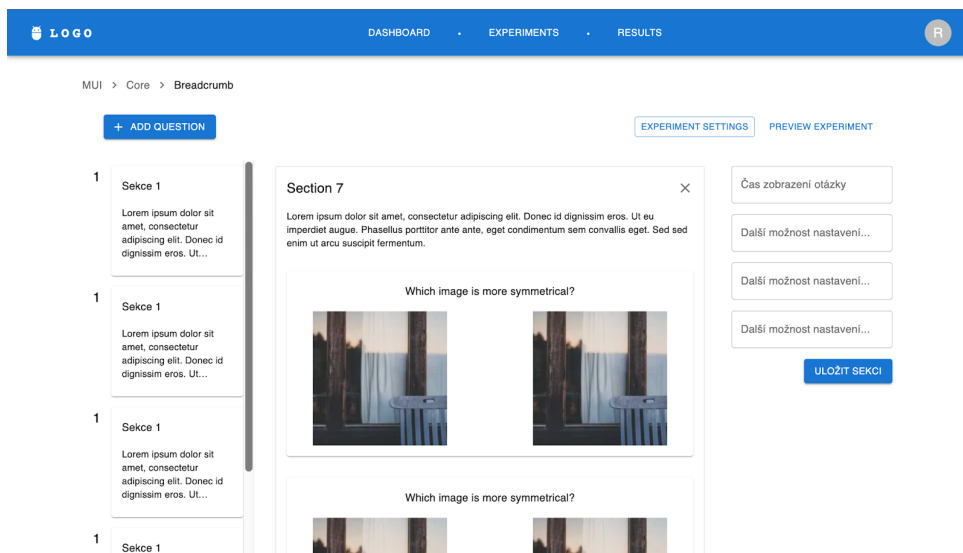


Figure 4.10: Design iteration 3 of section detail. The overall design also stays the same as in the previous iteration, but now is designed using HTML with the use of MUI tools instead of just sketches on paper.

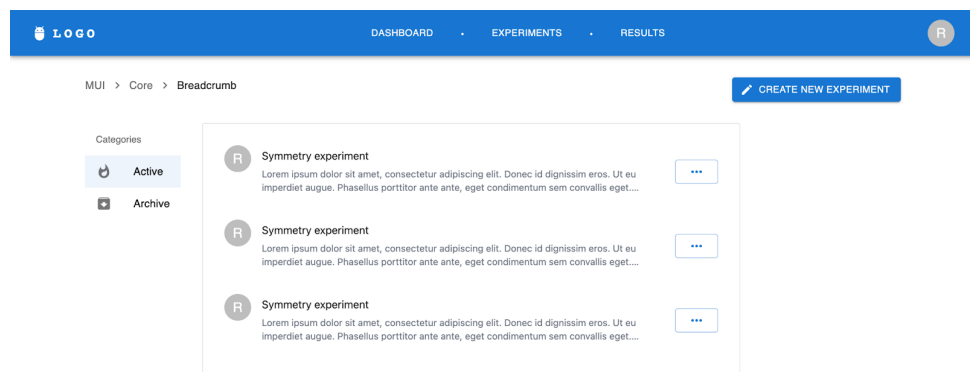


Figure 4.11: Design iteration 4 of experiment list. Final design of the experiment list page for this iteration implemented in HTML with the use of MUI tools. All further changes has been included in the very final design in section 7.1.

Chapter 5

Implementation

The design proposal and decisions are discussed in chapter 4. I will discuss and describe the technologies used during the implementation itself. Also, I will discuss code structure, implementation decisions, and a few unused possibilities.

5.1 Technologies Used in the Application

5.1.1 React

React [20] is an open source JavaScript library that is mainly used to build large scale single-page applications (SPAs). It was developed by Facebook (currently Meta - the company itself has been renamed) back in 2011, nowadays maintained by Meta, individual developers, many other companies ([1]), and is currently the most used web framework [31]. It is known for its performance, scalability, and reusability [1].

The key features, the three main pillars of React framework are:

- Component-based architecture
- Virtual DOM
- JSX

Component-Based Architecture

In React, the application UI can be broken down into very small parts, the so-called components. Each component is self-contained, has its specific functionality and properties, and it can have its state. Then, the application UI is made up of several of these components. Each component can be used many times in the application. If a change is made to one of the components, no other component is hit by the change. This ensures better maintainability, decreases the chance of error, and reduces the recurring code across the project.

■ 5.1.2 Next.js

In section 5.1.1 I mentioned that many higher-level frameworks are built on top of React. One such framework is Next.js [41], a full-stack framework designed to build Server Side Rendered (SSR) React applications. In comparison to pure React, Next.js provides us with a way to improve performance and SEO ranking (Search Engine Optimization) by pre-rendering pages on the server and sending them to a client partly or fully rendered.

The framework offers a technique to pre-render only parts of a page and fetch data on demand as well as automatic code splitting. Automatic code splitting is a mechanism aimed at reducing bundle sizes sent to the user by splitting the bundle into smaller chunks and importing the chunks dynamically, thus again improving performance.

As Next.js is a full-stack framework, by design, it provides a mechanism for routing, as well as a simple solution to build an API.

Key Advantages of Next.js.

- Server-side rendered pages -> better performance, much better SEO
- Excellent load times because of automatic code splitting
- Shallow learning curve for users familiar with React applications, even easier for users with experience with some kind of Node.js API and React routing
- Fast time to market
- Static site generation
- Wide community, many tutorials and blog posts with possible solutions and tons of out-of-the-box features

■ Why to choose Next.js?

I decided to choose the Next.js framework mainly because it is based on React. It can also be said that Next.js may be considered as a kind of React extension that provides server-side rendering, static site generation, routing, and a solution to build an API fast. Compared to Nest.js, which is another popular full-stack solution, Next.js is much more minimalistic, has a shallower learning curve, and a faster time-to-market.

■ 5.1.3 TypeScript

JavaScript is hands down the most popular programming language [31], but the developers' relationship with pure vanilla JavaScript is often love-hate. With the increasing size of a project, it becomes difficult to handle and be aware of the flow of the data and all the data types and easy-to-fix, but significant errors come to light.

■ NoSQL Databases

NoSQL databases, on the other hand, do not use any database schema. They can either employ a key-value schema, save documents as simple JSON objects, or save data in graph structures. Considering databases using JSON objects, such as MongoDB, they use collections as a NoSQL alternative to tables, and documents, which are JSON objects. Collections do not have any schema and are created automatically with the first document in a collection.

When a user wants to save a completely different document to a collection, he is free to do so. This creates a large margin for errors, but during the first stages of development, when a developer does not have a rigid data schema or is just playing around with different approaches, it can be very useful and time-saving.

■ Why to choose MongoDB?

When evaluating our requirements and needs, SQL databases come out as the best option using a rigid data schema. Our data, such as experiment details and settings, user details, or experiment results, have some kind of predefined data schema, so what is the reason I did not end up using the SQL database?

NoSQL could provide us with an easy way to alter the results schemas on the go. In the end, I optimized the time efficiency of the developer and opted for the hybrid option, the MongoDB NoSQL database [22], but with a MongooseJS [16] package providing a schema-based solution including validation. This gives us the freedom to make easy changes to document types during development phases, but with all the advantages of a rigid type SQL solution. Why specifically MongoDB? It is the most used NoSQL database [31] with great performance, easy local setup, easy cloud setup, and powerful command-line interface.

■ 5.1.5 Other Technologies Used in the Application

■ Cloudinary

Managing, uploading, delivering, and editing assets can be a challenging task in all web-based applications. For that part, I decided to use Cloudinary [5] instead of using a common database and saving the assets in binary format. Cloudinary is a free to use (to some scale) service providing a cloud-based image and video platform with nice-to-use API and an option to make edits and transformations to images directly inside the Cloudinary platform. Cloudinary provides the application with cloud storage for assets, image delivery, and the option to image processing, such as cropping, resizing, rotations, or format conversion.

Regarding security, although Cloudinary is closed source, it is an enterprise level tool that is trusted by a number of companies worldwide [6]. The application do not store any participant's data in the Cloudinary cloud

community of developers engages a heated debate over who defined and implements the best "best practices" and whether there is a good and a wrong way to something. Every developer has its own coding style, and good developers are able to vindicate their decisions and approaches. My coding style is based on multiple discussions and a few years of experience. Over time, my specific style keeps changing a little, but during the thesis implementation, I did not make any significant changes in the project structure, the coding style, etc.

I kept the structure of the whole project as simple as possible. Starting with the 'pages' folder, where the routed page components are stored. Each part of the application, each topic, is stored in a named folder, usually followed by an 'index.tsx' file. The React components used in these pages are stored separately in a global 'components' folder with preserved naming. For example, the 'experiments' page contains components stored in 'components/experiments'. I have divided each part of a page into separate folders so that a single component is responsible for the smallest possible chunk. This approach to file organization is sometimes called a component-centric structure. Also, any components that are used in more than one place in the application are stored separately.

■ 5.3.1 Stateful versus Stateless Components

When it comes to handling the application state, another good practice is to keep the state as low as possible in the component tree. I have stuck to keeping the state at a reasonably low level, such as the main component in the component-centric structure. It is not a good convention to let some low-level components handle state updates. As an example, consider a page that contains a list (a single component) of some editable elements as the main content. The lowest level to keep the state of the items and the changes is in the list component itself. But if the list is the main and only content of the page, it is better to keep this state at the page level and leave the list component stateless.

■ 5.3.2 React Hooks

React hooks and the exclusive use of functional components are a somewhat recent addition to React development. I decided to pull all the fetching logic out of the components and into separate hooks. This makes both the components and the logic much easier to maintain and debug. Using this technique, I was able to make the display components stateless, even when they needed to fetch data from the server.

■ 5.4 Code Quality

The project should be maintainable in the future and may be extended further. Good code style is necessary along with the use of some code quality tools.

As a code formatter, I chose Prettier [18], one of my favorite formatters with easy setup and very readable code output. Together with another tool, ESLint [44], they statically analyze the code and return errors and warnings according to their settings.

■ 5.5 Chapter Summary

In this chapter, I have introduced and described in detail the technologies and frameworks used for the purpose of implementation of the application (section 5.1). I have described the structure of the project in section 5.2 and covered the hot topic of best practices in both general and specifically for this project in a development process. I also presented tools that were used to ensure good readability of the application code and maintainability for future support.

Chapter 6

Testing the Application

Upon completion of the implementation of the application, I performed usability tests with people as close to potential users as possible.

Please note that both the description of the usability testing and the results of it is bounded to the application design. If you are not yet familiar with the application design and its functionality, head to the section 7.1 first, where I described the application in more detail.

6.1 First Round of Usability Testing

6.1.1 Background Summary

I have tested the first version of the final web application. Note that no notifications have been implemented yet and there were still some bugs (most of which were found during usability testing) and unimplemented functionality. I conducted the test remotely using TeamViewer [39] and a locally hosted application. I also screen-recorded the whole process for further analysis.

6.1.2 Methodology

The participant invited for the usability test was a former student in the same field of study as me, a developer himself. He was already familiar with usability testing and had been on both the researcher's and the participant's side. First, I put the participant in the picture with the information about the symmetry research and the information about 2-AFC tasks. Then I introduced the goal of the application to the participant and roughly explained the scenario in which he was working. I tried to mimic the real usage of the application, including the prerequisites and knowledge of the researchers who should use the application. I also conducted the usability test as a whole, mimicking real work with the web application.

I had four broadly specified test cases:

1. You have a bunch of images that you want to rank using the 2-AFC testing method. Create your user account and then create the user testing experiment (with questions of any type).

2. You have 8 images here. You want to compare 3 of them in an "each-with-each" way, and then compare the rest using your manual comparison settings with an additional text between the manual comparisons.
3. In the first experiment, you know that you want to have 3 separate groups of participants, where each group gets the questions in a different order. Set the experiment correctly and invite a few participants.
4. Play a participant role and execute the experiment you have created.

During the usability test, I altered the test cases and linked them together. I asked the participant to create an experiment. I took notes of his reactions, approaches, and feedback. When he was done with completing the experiment, I asked him to run the second test case, and so on. The last test case was mainly designed to provide feedback on the overall user experience and to check whether the execution of the experiment seemed natural to him according to the experiment creation.

■ 6.1.3 Results

The participant was able to complete all test cases without any significant problems. He paused twice during the study. He wanted to make sure he understood the 2-AFC task and the experiments correctly. Once during the first task, when he was not sure what the introduction section should mean in the final experiment. The other time was during the second task. The participant started to create the comparisons manually using the "blank section". After a while he hesitated and asked if he is right that the 2-AFC question compares the images in the "each-with-each" way and rearranged the experiment. Both times, he finally succeeded without my help, only with a repeated explanation.

■ 6.1.4 First Test Round Feedback

Upon starting the usability test with a login form, the participant liked the straightforward process and the user interface and randomly tried invalid inputs. The participant was asked to create an experiment. After saving, he paused for a second when he was returned to the list of all experiments. He told me: "I was expecting to be redirected directly to the newly created experiment. It was not clear to him at first glance that the newly created experiment was in the list of experiments and that he should click on it. Therefore, I recommend redirecting the user directly to the details of the experiment when it is created.

Another important feedback was related to the locking of experiments. The participant was unable to understand the concept of locking the experiment on his own. He told me that at first he thought that locking the experiment actually meant "stopping" it. Disabling the ability to invite participants, disallowing completion, etc. I recommend either renaming the "lock experiment"

or creating some kind of application onboarding to explain all the terms to the user.

Related to the locking mechanism, the participant also mentioned that instead of having two categories "active" and "archived", he would appreciate categories "locked" and "unlocked".

In terms of application design, the participant mentioned that he was missing some kind of integrity in the design. For example, on the experiment list page, he would like to see "something to fill in the blank space". Unfortunately, creating more white space to evoke a feeling of not being overwhelmed with information was a design choice based on the recommendations in the book [43].

6.2 Second Round of Usability Testing

6.2.1 Background Summary

The second round of usability testing was performed on the altered version of the application. Using the feedback and knowledge from the first round, I fixed the application bugs and implemented a few new features not yet available in the first version, like snackbar feedbacks. As in the first round, the usability test was carried out online using the TeamViewer [39] application and running the application locally on my machine.

6.2.2 Methodology

The second participant was once again a former student in the same field as I am. A young game designer and game developer with experience conducting usability tests. With no problems in organizing the first round of tests, I decided not to change either the test introduction or the tasks. The main goal of this usability test round was to collect bugs that were not discovered during the first usability test and to get feedback on the general ease of use of the application user interface.

6.2.3 Results

The second participant was also able to complete all specified tasks without problems. However, there was a minor problem when describing the symmetry experiments and the 2-AFC task to the participant. I was not clear enough about the 2-AFC task specification. The participant stopped at one moment when creating the 2-AFC question and asked me to repeat the specification and whether he understood it correctly.

Using repeated information about 2-AFC, the participant was able to complete all tasks. When asked to invite participants to the created experiment (still unlocked experiment), the participant was not sure whether the "lock experiment" is the right way to go. He hesitated for a moment, asked if it is the correct process, and without my help, he continued and tried to lock the

experiment on his own. When dealing with a locked experiment, there were no further problems.

6.2.4 Second Test Round Feedback

The registration, log-in, and creation of an experiment was very straightforward for the second participant. The participant appreciated the simple look of the log-in/register form and the modern clean look of the overall UI after logging in.

Although he liked the overall clean look, he mentioned that he "lacks something" to fill in the blank space. The feeling of "too much space" may have been partly caused by the large screen the participant used. He proposed to make the UI smaller and place it in the center of the screen, not to stretch it to the current width. Because the application is meant to be used on many different devices and on many different screen sizes, but mainly on devices with larger screen, the blank space on the large screen has its justification (we can read more about the usage of blank space in the application user interface in [43]).

The participant also appreciated the Material Design snackbar feedback and the experiment list on the main experiment page. He said the list is clear, well spaced, and the "locked" and "unlocked" categorization inside the single list, as seen in fig. 6.1, makes sense and is easy to understand.

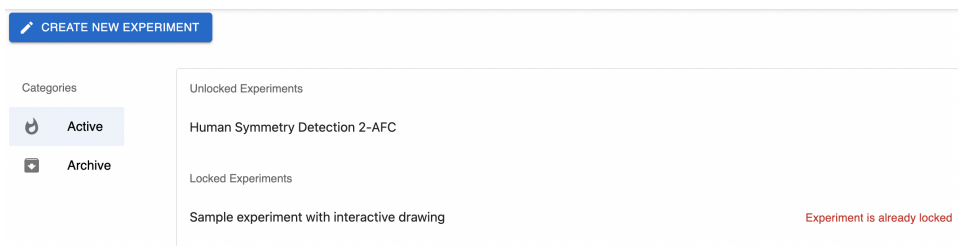


Figure 6.1: The list of experiments divided into locked and unlocked subcategory

The second participant marked the experiment locking mechanism as understandable and clear, but he added that the description of the locking mechanism should be explained somewhere during the onboarding process or that it should be renamed. It could have been caused by the starting instructions, where I highlighted that the experiment cannot be edited when a first invite to the experiment is sent.

6.3 Third Round of Usability Testing

6.3.1 Background Summary

After the second usability test, I fixed the bugs found during the second round and conducted the usability test on the altered application version. Unfortunately, I was unable to add a variety to the user testing environment,

and the third usability test was again conducted online using the TeamViewer [39] application.

6.3.2 Methodology

The third participant was a true potential user of the application. It was a Ph.D. student and a researcher himself with the main focus on data analysis and data visualization. Due to the participant's previous encounters with 2-AFC tasks, I decided not to provide him with the explanation of 2-AFC psychophysics experiments and observe his orientation across the UI and his application flow.

Regarding the test cases, I slightly altered them during the testing process. I moved the third task dealing with the participant groups to the first task. Also, since the third participant has experience with visualization and data analysis, I decided to include tasks regarding the analysis and download of experiment results. Therefore, the final tasks were as follows:

1. You have a bunch of images that you want to rank using the 2-AFC testing method. Create your user account and then create the user testing experiment with some introduction, main 2-AFC and/or other parts and some final words. Set the experiment for 3 separate groups of participants and invite a few participants.
2. You have 8 images here. You want to compare 3 of them in an "each-with-each" way, and then compare the rest using your manual comparison settings with an additional text between the manual comparisons.
3. Play a participant role and execute the experiment you have created.
4. View the results of the previously filled in experiment and download the CSV export of the results. Analyze the CSV file.

6.3.3 Results

The third participant dealt with the tasks assigned the best. He was able to fulfill all the test cases along with providing me with feedback. The only problematic part was regarding the locking experiment mechanism. He explained to me that it was clear to him that something like "finish the experiment" is necessary before inviting the participants to the experiment, but at first glance it was not obvious that the "lock experiment" button is exactly what he was looking for. Other than that, the participant had no problem completing all the tasks and overall was well oriented in the UI.

6.3.4 Third Test Round Feedback

From the start, the participant highlighted the easy-to-use and clear UI of both the login screen and the main application screen. It was obvious that the participant had previous experience with Material Design UIs, because

he immediately recognized and described the primary and secondary options and knew where to navigate to achieve the assigned task.

The process of creating a new experiment was flawless and the participant took a kind of trial-and-error approach. During the creation, he discovered and described to me aloud what he had seen and how he had understood it. He described the different types of sections and intentionally tried to add them and delete them afterward. Without any further instructions, he created an experiment containing an introduction with a custom text, 2-AFC tasks, and a final word with acknowledgment. He also set the number of groups of participants and played with different experiment settings. He mentioned a bug with blank section settings, where "you can set image sizes and gap between images even when there is no question with images in the section". Regarding the section settings, he also mentioned missing units in one of the options, when there were specified units in the other options. When moving to the second task, after the task assignment he immediately described the process needed to complete the case. I did not want to interfere and let the participant go through the process. He described the process correctly and had no problem accomplishing the task.

Moving to the third task, some irregularities from the creation process arisen. When the participant moved to the "public participant's view of the experiment", he was a bit confused by the previously set question titles and the section titles and descriptions. As for the first created experiment, it was not clear to him that the section title and section description are invisible in the experiment completion process, and only the question titles are the real what-to-do instructions for the experiment participant. After he took a cursory look around, he understood the concept and went back to the creation of the experiment. With no instructions he copied the previously created experiment and edited it the way he had wanted. During this process, the participant found a bug in the application, which caused that the copied experiment had different order of sections than the original one.

Apart from the assigned task, the participant played around with the experiment edit mode and tried to copy a few experiments, create, edit, and delete different sections, and change the order of sections. He tried to drag-and-drop the section cards to change the order, which was a dead end. He mentioned that to change for example a second section type, a researcher needs to delete all but one section and re-add (together with re-setting and re-uploading the images) them later. This is definitely a major flaw that needs to be addressed, and drag-and-drop should be added to the section cards. The participant also brought up a question about the permutation selection for the different groups of participants in the experiment. In the current version of the application, I selected random permutations for the different groups. He suggested to use something to ensure a larger distance between individual question permutations. This problem will be a subject of further discussion, but it is not certain to change the permutation selection.

The final task, prepared specifically for this participant, was again seamless. The results page is, according to the participant, clear, and the filtering of

the results is intuitive. The download process is straightforward. Looking at the resulting CSV, the participant rated the columns as well-assembled, but with pretty big overhead with multiple columns containing data regarding the whole experiment, and thus the same data for all the rows. We discussed the possibilities of moving the data to some kind of metadata. After brief participant's insight about the problems with missing metadata in many CSV files he had been dealing with, I decided to leave the CSV formatting without changes.

Overall, the participant appreciated the familiar and easy-to-use Material Design and the overall design of the application.

6.4 Findings and Changes in the Application

I have described the specific results of each round of user testing in sections 6.1.4, 6.2.4 and 6.3.4, let us now look at the overview of possible modifications based on the usability testing feedback.

The most important finding from the conducted usability tests is definitely the "Lock experiment" process. All three participants had some problem understanding what to do when the experiment creation is finished and they want to invite participants to it. I recommend to implement some kind of an onboarding process, where the user will get familiar with all the processes and settings in the application. The onboarding may be of different types, such as a comprehensive process on the first login to the application. It can also be divided into specific parts. If the user, for example, navigates to the experiment creation, the onboarding process specifically for that page would run, etc.

Another significant finding was the lack of changing the order of sections and questions. Without the possibility to change the order, the user would have to delete all following sections if he had wanted to add a new section somewhere in the middle. Since I use Material Design cards to display the sections and question in the application, the most user-friendly solution is to add drag-and-drop support. The third participant tried to change the order using the drag-and-drop approach and it was natural for him. Since the lack of ordering the sections was a serious finding degrading the application ease of use, as a result, I implemented ordering using drag-and-drop approach for both sections and questions (in the Blank section type).

6.5 Chapter Summary

I have conducted three independent usability tests with participants from different fields. During the first usability test, I have discovered some major bugs in the application and collected both positive and negative feedback. Having fixed the bugs, in the second round of usability test I have confirmed some irregularities raised during the first round, like the experiment locking mechanism. The third round was the most helpful. I have tested the ap-

plication with a participant who is a true potential future user of the app. And again, the third round verified that we should think over the experiment locking mechanism and discuss some not-yet-implemented functionality, like sections reorder.

All three rounds of usability testing confirmed that the design of the application, its functionality, and ease of use is on point. We have verified that the Material Design approach was a good step and the aim on user-first and as-simple-as-possible design is well applied.

Chapter 7

Results

This chapter describes the results obtained in this thesis and the current state of the application (section 7.1). Section 7.2 is dedicated to the discussion and evaluation of the requirements defined in section 3.1. In section 7.3 I will discuss the not yet implemented functionality that was raised during the analysis of state-of-the-art solutions and usability testing. Also, I will add some options for future extensions and the overall direction of the application.

7.1 Current State of the Final Application

Since this thesis' focus is not research regarding the results of the psychophysics experiment but to implement a tool to create such experiments, I will describe the application as a whole and provide a simple flow through the main functionality.

The final application can be divided into two strictly divided parts that share a codebase. The abstract separating line is an authentication gate seen in fig. 7.1 that researchers use to create and manage their experiments. All functionality designated for researchers is behind the authentication. The other part is public pages for participants to fill in experiments they are invited to. Every participant gets his own direct link bounded to his email only, where he can both complete the experiment or remove the consent to share his answers and thus delete his results in the experiment (as seen in fig. 7.7). Note that the participant is able to delete his answers at any time in the future by following the unique direct link, where he has completed the experiment.

7.1.1 Researchers' application

The main page of the researchers' application is the experiment list page (fig. 7.2). From this page, a user can create a new experiment or edit any previously created one.

In a single experiment editing mode, the experiment is divided into sections (fig. 7.3). Section can be either a predefined introduction and acknowledgment, or it can be of types 2-AFC and Blank. The Introduction and Acknowledgments sections contain only formatted text ready for editing. The Blank

Figure 7.1: Simple login page of the application dividing researchers' and participants' parts of the application

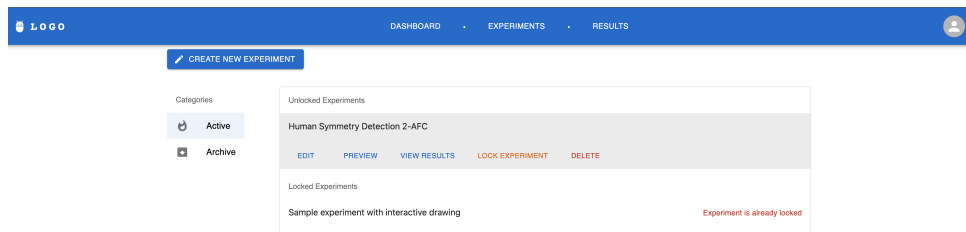


Figure 7.2: Main page with list of previously created experiments. The experiments are divided into two major categories - active and archive. In the active category, the experiments are divided into subcategories locked and unlocked, simplifying the orientation even more. Clicking an experiment opens a menu row with additional experiment actions.

section provides the user with the possibility to format the experiment manually. The user can then add questions of three types (Plain text, Image comparison, Interactive drawing question) to the Blank section (fig. 7.4). No other section type has the possibility to manually add questions. The 2-AFC section allows the user to upload a bunch of images. The application then handles the logic and generates all possible pair comparisons, which a participant filling in the experiment will answer. To summarize the experiment edit mode, there are 3 types of questions available, there is a possibility to add a plain text with for example instructions anywhere in the experiment and the user is able to set and modify the order of individual questions.

After completing the creation of an experiment, the user can lock the experiment. This disables the editing mode of the experiment, and the user can invite participants using their email address. On the results page, the user can export experiment results in CSV format (fig. 7.5).

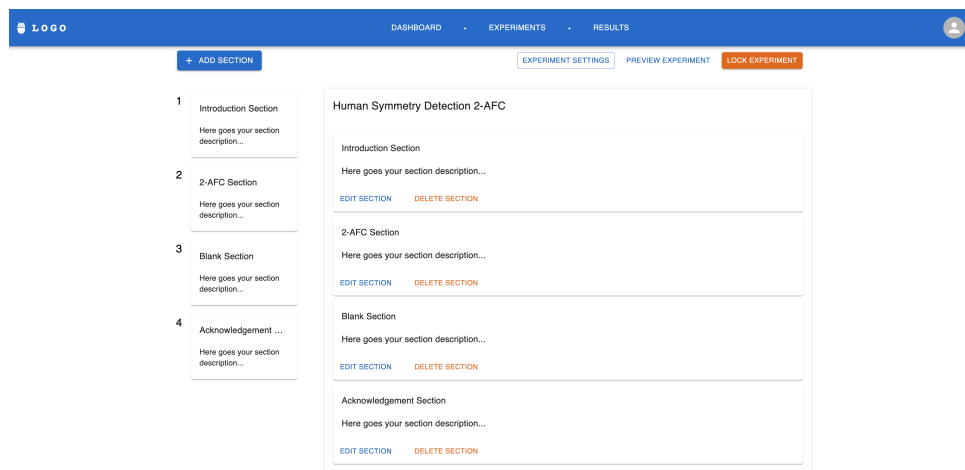


Figure 7.3: Detail of an experiment that contains one introduction section, a 2-AFC section, a blank section and an acknowledgement section. The order of sections can be changed using drag-and-drop approach.

7.1.2 Participants' application

The participants' application is much simpler. After inviting the participant using his email, the participant gets his unique direct link to complete the experiment. He can see only the question headers (we can see one example question in fig. 7.6) and fills in the experiment question after question as the researcher has set the experiment.

After completion of the whole experiment, the participant is able to withdraw his consent to save his answers (fig. 7.7). By doing so, he remotely deletes all the results collected from him in that experiment. He can access the page to withdraw his consent any time in the future using the direct link to the experiment completion.

7.2 Meeting the Requirements

As described above in section 7.1, the application requirements defined in chapter 3 have been met. Although I decided to develop a single application for both researchers and participants, these two parts of the application are strictly divided and participants are not able to access anything that was not meant to them.

Researchers are able to create 2-AFC experiments, set exact question types, and exact order of the questions. They can also add an interactive question, such as drawing a line into an image, or add a plain text question with e.g. instructions before a section focusing on a different goal.

After completing an experiment by some participants, the researcher is able to export and download the experiment results in CSV format, allowing further analysis of each individual question answer.

In terms of the participants' application, participants can access the ex-

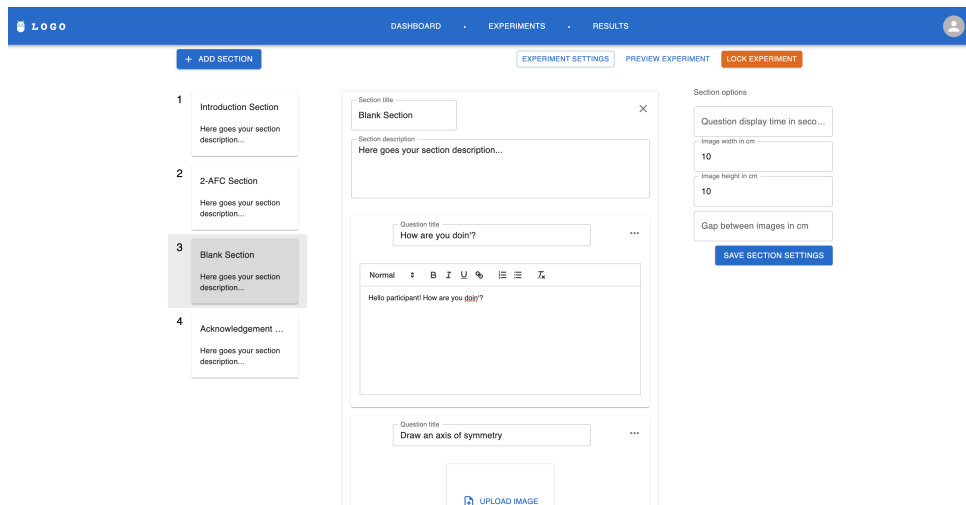


Figure 7.4: Detail of a blank section containing a single plain text question and an interactive drawing question. The order of question within the section can be changed using drag-and-drop approach.

periments they were invited to and fill them in. They are also allowed to withdraw their consent with data sharing and delete all their collected data (answers) in an experiment.

Talking about non-functional requirements (section 3.1.2), the application was developed to be used primarily on a computer with larger screens. Privacy-wise, the participants' answers are only bound to their email address. Also, a user of the application has no way to find out the connection between an individual participant and his answers. The only connection is being used on the database level for the participant invitations and the links for experiment completion to work. Based on the results of the usability test summarized in section 6.4, I can responsibly say that the final application is easy to use, meets its purpose, and has a meaningful user experience.

7.3 Further Extension

The implementation of the application reached its end during this thesis, meeting all the requirements previously set and including the conducted usability testing. There are, however, a few ideas and options to consider for future work on the app.

As suggested in section 6.4, some onboarding process with the detailed description of procedures and options in the application could be beneficial because of the complexity of the application. Unfortunately, there was not enough time to implement such solution. The onboarding is thus added as a possible further extension with greater priority.

From the start, the application was written with extensibility and sustainability in mind. Thus, adding a new type of section or a new type of question is an easy task and does not take the developer more than a few

ID ▼	Experiment Name	# of Respo...	# of Groups	Export to CSV
646f66cc5...	Human Symmetry Detection	5	3	GENERATE CSV

Rows per page: 100 ▼ 1-1 of 1 < >

Figure 7.5: Detail of experiments results table containing results for a single experiment (filtered by ID - depicted by the filter icon in the column header). There is an option to filter based on each column, sort based on different columns or hide the columns entirely.

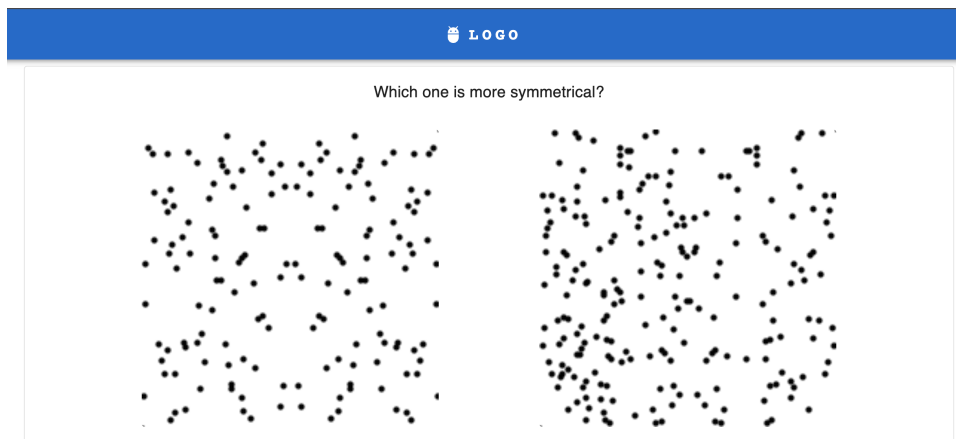


Figure 7.6: Participant's view of a 2-AFC comparison question during the experiment completion. Dimensions of the images is set by the researcher during the creation.

hours. An extensibility like that adds the application huge amount of utility and prepares the application for turning into a big project, having Qualtrics

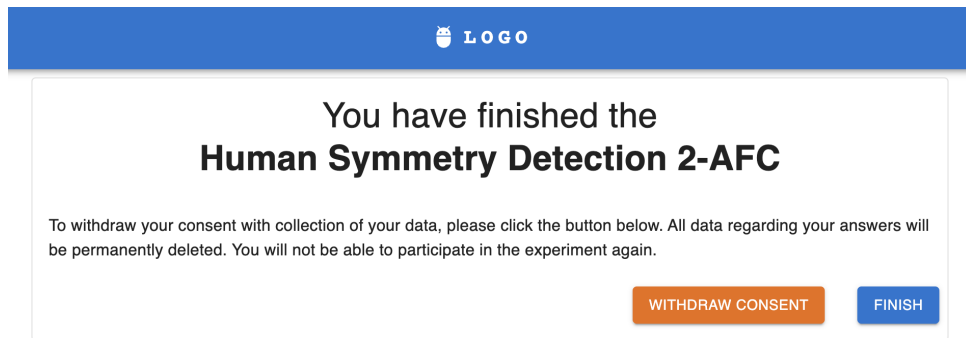


Figure 7.7: Final page of the participant's experiment completion with the option to withdraw his consent with saving results. Clicking the withdraw consent button, the application deletes all data collected during the experiment completion.

(section 3.2.2) in mind.

An integration with Amazon Mechanical Turk, mentioned during the analysis of Versus (section 3.2.1) could be another beneficial addition. The current app version aims at all-manual experiment settings, including the selection and invitation of specific participants. The MTurk integration may open doors to less specific quantitative research, where the amount of results needed exceeds the options of manual settings by a single person.

Moving from the "experiment creation" part, the application could implement some visual dashboard to briefly analyze the data before moving to the detailed analysis using the CSV exports. For the purpose of this thesis, I did not design features for the analysis part and focused only on the administration part.

7.4 Chapter Summary

We have taken a look at the current state of the final application and a simple application flow in section 7.1. Using the flow and usage of the application, we have resolved that the requirements set in chapter 3 were met. Also, some options for further extension and support of the application have been presented in section 7.3 with interesting ideas for a wider application usage.



Chapter 8

Conclusion

The objective of this thesis was first to analyze psychophysical experiments, similar experiment questions not directly belonging to psychophysics, and state-of-the-art tools used for administration and completion of such experiments. Subsequently, the essential objective was to design and implement a solution based on the predefined requirements.

There are multiple tools for the creation and management of psychophysical and similar experiments, but none of them met all the requirements. The existing tools are only frameworks, and the user needs some coding skills to be able to utilize their potential or they are too complex and not really easy to use. I analyzed and tested the existing solutions and, on the basis of that, started to design my solution for the problem. After defining the requirements, I started with some ideas and proposals on the division of experiments using simple sketches. After that, I selected an appropriate technology stack for the implementation and followed with the implementation design. I focused mainly on a modern look, appropriate functionality, ease of use, and the future extendability.

Based on the design, I developed a fully functional web application for creation, administration, and completion of 2-AFC experiments with the support of interactive questions. I described the design and implementation in more detail including the technological stack that consists of technologies like Next.js, React, TypeScript or MongoDB.

I also tested the application using the usability testing approach on three participants, processed the feedback, and proposed and implemented changes to the discovered problems.

I described the current state of the application and its possible future, including a few ideas for future extensions.

The application will serve as a tool used by the research team at the Department of Computer Graphics and Interaction, CTU, and I will be at disposal for future application support.

The thesis also provided me with a lesson regarding the development process as a whole, from requirements mining and processing, through the design process and planning the technological stack to the implementation, user testing and feedback handling.

Appendix A

Application Manual

A.1 Create and Configure an Experiment

1. Open the application with logged-in user.
2. Click the "Experiments" menu button in the top menu bar (skip this step if you already see the list of experiments or if you see the text "No experiments").
3. Click the button "Create new experiment" in the top left corner and fill in the experiment details. The same button is also in the center of the screen, when there are no experiments created yet.
4. The newly created experiment appeared in the experiment list. Click the experiment list item and click the "Edit" button in the sub-row.
5. To edit the experiment title and description, click the "Experiment settings" button in the top right corner. In the experiment settings, you can also alter the number of participant groups. Participants in different groups get the 2-AFC comparison questions in different order.

A.2 Setup the Content of the Experiment

1. Head to the experiment edit page (see appendix A.1).
2. Click the "Add section" button in the top left corner. Here you can choose the type of section that you want to add.
 - Introduction section
Plain text section with predefined content, where you can provide the participant with some introductory words.
 - 2-AFC section
Section, where you upload multiple images and the participant chooses exactly one of two choices (comparison of tuples from the collection of the images). The application handles the logic behind the generation of all image comparisons in each-with-each way.

- Blank section

Blank section supports multiple different question types:

- Plain text

Editable formatted plain text with the continue button.

- Select one of two images

A single 2-images comparison where you manually set both images.

- Interactively draw a line in the image

A single image with drawing interaction. During the experiment, the participant can draw a line in the image.

- Acknowledgement

Plain text section with predefined acknowledgement words. As in the Introduction section, you can edit the text content of this section.

A.3 Finish Experiment Creation and Invite Participants

1. When you are finished with the experiment setup in appendix A.2, click the "Lock experiment" button.
 - If you are on the page with the list of experiments, the "Lock experiment" button is located in the sub-row visible after clicking the chosen experiment.
 - If you are inside the experiment on the experiment detail page, the "Lock experiment" button is located in the top right corner.
2. The "Lock experiment" will prevent all further changes to the experiment and will make the option to invite participants available.
3. Invite participants to participate in the experiment.
 - If you are on the page with the list of experiments, the "Manage participants" button is located in the sub-row visible after clicking the chosen experiment.
 - If you are inside the experiment on the experiment detail page, the "Manage participants" button is located in the top right corner.
4. In this modal, you can invite new participants using their email address. After adding the participant to the list, you can send him an email invitation using the icon button on the left side (with mouse hovering on the chosen participant list item). The "trash" icon button will delete the participant from the experiment and prevent him from filling out the experiment.



Appendix B

Master Thesis Specification

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Jandek** Jméno: **Martin** Osobní číslo: **475389**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávací katedra/ústav: **Katedra počítačové grafiky a interakce**
Studijní program: **Otevřená informatika**
Specializace: **Interakce člověka s počítačem**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Aplikace pro experimenty zkoumající symetrii v obrázcích

Název diplomové práce anglicky:

Applications for experiments exploring symmetry features in images

Pokyny pro vypracování:

Analyzujte způsoby provádění a vyhodnocování uživatelských experimentů, které se zaměřují na kvantitativní hodnocení vlastností obrázků nebo krátkých animací. Zaměřte se na testy využívající Two-Alternative-Forced-Choice (2AFC). Dále nastavujte problematiku vyhodnocování experimentů zkoumajících symetrii, viz např. [1] a [2].

Na základě analýzy navrhnete dvě webové aplikace, administrátorskou aplikaci pro tvorbu jednotlivých testů a jejich vyhodnocování a klientskou aplikaci pro účastníky experimentů, která bude sloužit k přihlášení na experiment a jeho provedení. Administrátorská aplikace bude umožňovat víceuživatelský přístup a správu jednotlivých experimentů, uživatelů a účastníků experimentů. Dále bude aplikace umět nastavit, jaké typy úloh budou součástí experimentu a jakým způsobem budou jednotlivé úlohy řazeny, aby byl výsledek testu validní. Kromě implementace standardních variant 2AFC testu vytvořte také interaktivní variantu úlohy, která umožní interaktivní vkládání os symetrie do obrázku.

Na základě návrhu naimplementujte administrátorskou a klientskou webovou aplikaci s responzivním designem pro větší i menší displeje.

Výsledné aplikace otestujte kvalitativními uživatelskými testy na datech dodaných vedoucím práce, nejméně na 3 sadách experimentálních dat.

Seznam doporučené literatury:

[1] Chaehan So (2022) Measuring Aesthetic Preferences of Neural Style Transfer: More Precision With the Two-Alternative-Forced-Choice Task, International Journal of Human-Computer Interaction, DOI: 10.1080/10447318.2022.2049081

[2] Anděl Ondřej, Hledání symetrie v geodatech, Diplomová práce, Západočeská univerzita v Plzni, 2022, <http://hdl.handle.net/11025/49086>

[3] Bučkovský Dmitrij, UX Eval - Webová aplikace pro analýzu testů použitelnosti, Diplomová práce, České vysoké učení technické v Praze, 2018, <https://dspace.cvut.cz/handle/10467/77068>

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Ivo Malý, Ph.D. katedra počítačové grafiky a interakce FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **06.09.2022**

Termín odevzdání diplomové práce: **26.05.2023**

Platnost zadání diplomové práce: **19.02.2024**

Ing. Ivo Malý, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Appendix C

Bibliography

- [1] S. Aggarwal *et al.*, “Modern web-development using reactjs,” *International Journal of Recent Research Aspects*, vol. 5, no. 1, pp. 133–137, 2018.
- [2] S. Baliotti, “nodegame: Real-time, synchronous, online experiments in the browser,” *Behavior Research Methods*, vol. 49, no. 5, pp. 1696–1715, Oct 2017. Available: <https://doi.org/10.3758/s13428-016-0824-z>
- [3] D. H. Brainard and Spatial Vision, “The psychophysics toolbox,” *Spatial vision*, vol. 10, no. 4, pp. 433–436, 1997.
- [4] S. Chaehan, “Measuring aesthetic preferences of neural style transfer: More precision with the two-alternative-forced-choice task,” *International Journal of Human–Computer Interaction*, vol. 39, no. 4, pp. 755–775, 2023. Available: <https://doi.org/10.1080/10447318.2022.2049081>
- [5] Cloudinary Ltd., “Cloudinary,” online, 2012. Available: <https://cloudinary.com/> Accessed: 2023-05-08.
- [6] —, “Cloudinary customers,” online, 2012. Available: <https://cloudinary.com/customers> Accessed: 2023-05-08.
- [7] J. W. Creswell and J. D. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*. Thousand Oaks, CA: SAGE Publications, 2018.
- [8] J. R. de Leeuw, R. A. Gilbert, and B. Luchterhandt, “jspsych,” online, 2023. Available: <https://www.jspsych.org/7.3/> Accessed: 2023-03-06.
- [9] —, “jspsych: Enabling an open-source collaborative ecosystem of behavioral experiments,” *Journal of Open Source Software*, vol. 8, no. 85, p. 5351, 2023. Available: <https://doi.org/10.21105/joss.05351>
- [10] J. R. de Leeuw and B. A. Motz, “Psychophysics in a web browser? comparing response times collected with JavaScript and psychophysics toolbox in a visual search task,” *Behav Res Methods*, vol. 48, no. 1, pp. 1–12, Mar. 2016.

- [11] Google LLC, “Material design,” online, 2014. Available: <https://m3.material.io/> Accessed: 2023-05-08.
- [12] —, “Material design - cards,” online, 2014. Available: <https://m3.material.io/components/cards/overview> Accessed: 2023-05-08.
- [13] B. E. Hilbig, “Reaction time effects in lab- versus web-based research: Experimental evidence,” *Behavior Research Methods*, vol. 48, no. 4, pp. 1718–1724, Dec 2016. Available: <https://doi.org/10.3758/s13428-015-0678-9>
- [14] S. Kavanova, “Generátor symetrických dat s různými poškozeními symetrie,” online, 2023. Available: <https://github.com/StepkaK/LentilsGenerator> Accessed: 2023-03-22.
- [15] M. Kleiner, D. Brainard, D. Pelli, C. Broussard, T. Wolf, and D. Niehorster, “Psychtoolbox-3,” online, 1997. Available: <http://psychtoolbox.org/> Accessed: 2022-10-10.
- [16] LearnBoost, “Mongoosejs,” online, 2010. Available: <https://mongoosejs.com/> Accessed: 2023-05-08.
- [17] J. Lin and I. V. Bajić, “A platform for subjective image quality evaluation on mobile devices,” in *2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, ser. 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), 2016, pp. 1–4. Available: <https://doi.org/10.1109/CCECE.2016.7726650>
- [18] J. Long, “Prettier,” online, 2017. Available: <https://prettier.io/> Accessed: 2023-05-08.
- [19] K. McCusker and S. Gunaydin, “Research using qualitative, quantitative or mixed methods and choice based on the research,” *Perfusion*, vol. 30, no. 7, pp. 537–542, 2015.
- [20] Meta Platforms, Inc., “React,” online, 2013. Available: <https://react.dev/> Accessed: 2023-05-08.
- [21] Microsoft Corporation, “Typescript,” online, 2012. Available: <https://www.typescriptlang.org/> Accessed: 2023-05-08.
- [22] MongoDB, Inc., “Mongodb,” online, 2009. Available: <https://www.mongodb.com/> Accessed: 2023-05-08.
- [23] Mozilla Foundation, “Document object model,” online, 2005. Available: https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model Accessed: 2023-05-08.
- [24] J. W. Peirce, “Psychopy,” online, 2002. Available: <https://psychopy.org/> Accessed: 2022-10-10.

- [25] —, “Psychopy—psychophysics software in python,” *Journal of Neuroscience Methods*, vol. 162, no. 1, pp. 8–13, 2007. Available: <https://www.sciencedirect.com/science/article/pii/S0165027006005772>
- [26] —, “Generating stimuli for neuroscience using psychopy,” *Frontiers in Neuroinformatics*, vol. 2, 2009, original Research. Available: <https://www.frontiersin.org/articles/10.3389/neuro.11.010.2008>
- [27] J. Peirce, J. R. Gray, S. Simpson, M. MacAskill, R. Höchenberger, H. Sogo, E. Kastman, and J. K. Lindeløv, “Psychopy2: Experiments in behavior made easy,” *Behavior Research Methods*, vol. 51, no. 1, pp. 195–203, Feb 2019. Available: <https://doi.org/10.3758/s13428-018-01193-y>
- [28] J. Peirce, R. Hirst, and M. MacAskill, *Building experiments in PsychoPy*. Sage, 2022.
- [29] Qualtrics, “Qualtrics,” online, 2017. Available: <https://www.qualtrics.com/> Accessed: 2022-10-10.
- [30] D. A. Silverstein and J. E. Farrell, “Efficient method for paired comparison,” *Journal of Electronic Imaging*, vol. 10, no. 2, pp. 394–398, Apr 2001. Available: <https://doi.org/10.1117/1.1344187>
- [31] Stack Exchange, Inc., “Stack overflow developer survey,” online, 2023. Available: <https://survey.stackoverflow.co/2022/#technology-most-popular-technologies> Accessed: 2023-05-08.
- [32] G. Stoet, “Psytoolkit,” online, 2010. Available: <https://www.psytoolkit.org/> Accessed: 2023-03-06.
- [33] —, “Psytoolkit: A software package for programming psychological experiments using linux,” *Behavior Research Methods*, vol. 42, no. 4, pp. 1096–1104, Nov 2010. Available: <https://doi.org/10.3758/BRM.42.4.1096>
- [34] —, “Psytoolkit: A novel web-based method for running online questionnaires and reaction-time experiments,” *Teaching of Psychology*, vol. 44, no. 1, pp. 24–31, 2017. Available: <https://doi.org/10.1177/0098628316677643>
- [35] S. Sukamolson, “Fundamentals of quantitative research,” *Language Institute Chulalongkorn University*, vol. 1, no. 3, pp. 1–20, 2007.
- [36] Tailwind Labs Inc., “Tailwind ui,” online, 2020. Available: <https://tailwindui.com/> Accessed: 2023-05-08.
- [37] —, “Tailwind ui settings screen,” online, 2020. Available: <https://tailwindui.com/components/application-ui/page-examples/settings-screens> Accessed: 2023-05-08.
- [38] O. Tassinari and M. Brookes, “Mui,” online, 2014. Available: <https://mui.com/> Accessed: 2023-05-08.

- [39] TeamViewer SE, “Teamviewer,” online, 2005. Available: <https://www.teamviewer.com/> Accessed: 2023-05-08.
- [40] R. Ulrich and J. Miller, “Threshold estimation in two-alternative forced-choice (2afc) tasks: The spearman-kärber method,” *Perception & Psychophysics*, vol. 66, no. 3, pp. 517–533, Apr 2004. Available: <https://doi.org/10.3758/BF03194898>
- [41] Vercel Inc., “Next.js,” online, 2016. Available: <https://nextjs.org/> Accessed: 2023-05-08.
- [42] J. Vuong, S. Kaur, J. Heinrich, B. K. Ho, C. J. Hammang, B. F. Baldi, and S. I. O’Donoghue, “Versus—a tool for evaluating visualizations and image quality using a 2afc methodology,” *Visual Informatics*, vol. 2, no. 4, pp. 225–234, Dec 2018. Available: <https://www.sciencedirect.com/science/article/pii/S2468502X18300433>
- [43] A. Wathan and S. Schoger, *Refactoring UI*. Adam Wathan & Steve Schoger, 2019. Available: <https://www.refactoringui.com/>
- [44] N. C. Zakas, “Eslint,” online, 2013. Available: <https://eslint.org/> Accessed: 2023-05-08.
- [45] M. Čadík, M. Wimmer, L. Neumann, and A. Artusi, “Evaluation of hdr tone mapping methods using essential perceptual attributes,” *Computers & Graphics*, vol. 32, no. 3, pp. 330–349, 2008. Available: <https://www.sciencedirect.com/science/article/pii/S0097849308000460>