Creation of a novel usability test and conducting it on music streaming applications

Bachelor Thesis
Faculty of Science, University of Bern

submitted by

Korab Bejta

from Bern, Switzerland

Supervision:

PD Dr. Kaspar Riesen
Institute of Computer Science (INF)
University of Bern, Switzerland

Abstract

In this thesis a usability test has been created and conducted on three prominent music streaming applications: Spotify, Apple Music and Deezer. The aim of the study is to evaluate the usability of these platforms through a comprehensive usability test, which applies eight usability principles. The thesis also aims to serve as a possible guide for creation of future usability tests. The research questions of this thesis focus on the creation of a tailored usability test, the performance of the applications under the test and the correspondence between the results of the usability test and global popularity of the applications. The usability test involved twelve carefully selected participants. Each participant completed a list of tasks and then rated their experience of using respective application in a retrospective interview. Descriptive statistics were applied to analyse the results, revealing Spotify and Apple Music as equally user-friendly with each showing the best results in four out of eight principles. There were minimal differences between the two, but both had significantly better scores than Deezer. The analysis of the correspondence with global popularity shows a match for the Deezer app. However, the difference in the popularity of the other two apps is not reflected in the results of the test. Limitations of the thesis include biases in selection of the participants.

Acknowledgements

I would like to thank Dr. Kaspar Riesen for letting me write this thesis in this field even though it is not the research field of his group. I am also thankful for his feedback and for the flexibility he has given me. I would also like to thank all the participants for their willingness to take part in the usability test.

Contents

Chapter 1

Introduction

1.1 Background

Human-computer interaction (HCI) is a multidisciplinary field whereas computer science for the most part focuses on algorithmic study. It connects human psychology with design and implementation of software. However, due to its fundamental connection to the hardware and software of computer systems, it is considered a subfield of computer science. If one disregards the constraints of HCI whilst designing an interactive computer system, the system may fail due to inadequate consideration of contextual factor of the user. A large area of HCI is dedicated to user interaction, which underlines the crucial need to consider the user and their task context. Taking user interaction into consideration whilst designing user interfaces (UI) helps in avoiding creating suboptimal interfaces and minimises risks to the computer system as a whole. [1]

One key subfield of HCI is the user experience (UX). User experience refers to the overall perception and interaction that individuals have with a computer system. This includes every aspect of the user's interaction. such as the user's feelings, emotions, preferences and perception. Embedded within the broader concept of UX is the UI. The UI refers specifically to the visual and interactive elements with which the user can interact in a computer system. User interfaces play a central role in shaping the overall UX. [2]

Developing and designing a tangible and accessible interface is called usability engineering. The primary goal of usability engineering is the optimization of the user-friendliness of the interface, which can be achieved by optimising efficiency effectiveness and the satisfaction of the user. These characteristics are often described as usability design principles that must be followed for good usability. These characteristics however, cannot be generalised and must be defined in each case. Each computer interface has different purposes and so there is no general way to measure them. As in all optimsation problems, the validation of each change is an important step. In usability engineering this step is called usability testing. . [3]

The cost-benefit analysis of usability engineering by Karat [4] shows the value of spending resources in usability engineering. These analyses in particular highlight the positive impact of systems that were designed with usability testing. The results of these cost-benefit analyses show significant benefits across different project sizes. The cost-benefit ratios range from 1:2 for smaller projects to 1:100 for larger projects.

1.2 Definition of the Problem

As already mentioned, usability testing is an important step in optimising the usability of an interface and can lead to significant benefits. Although usability testing has all these benefits, it is nevertheless largely done inadequately or completely neglected. [5]. The fact that there is no general usability test is the reason why usability testing is so costly and time-consuming. A smaller company that has to be careful with their budget would first have to define a user demographic of the interface. Then they must define usability characteristics and find out ways to test them in a suitable way. As this is a relatively lengthy process and the benefits are not immediately visible, smaller companies often neglect usability testing.

1.3 Aim of the Thesis

This thesis presents an example of a single-person usability test for music streaming applications. This is done to further investigate the question of how extensive the costs for all the steps of a usability test are. Each step of the creation and conduction of the usability test will be described in this thesis. The following thesis statement is explored: “Investigating the feasibility and efficacy of the creation and conduction of a single-person usability testing approach for music streaming applications” Another aim for this thesis is to show what to focus on in the creation and conducting of a usability test. This is done by describing the individual aspects of the test. It is intended as an aid to creating and conducting a usability test, as it shows what to focus on. Furthermore this thesis serves as a way to show developers how a usability test can lead to further improvements in their design and the overall usability.

1.4 Methodological Approach

This thesis focuses on usability testing, with a special focus on the creation and conduction of a usability test for globally used music streaming applications. The three music streaming applications that were chosen for this analysis are Spotify, Apple Music and Deezer.

In order to carry out the analysis, framework conditions and usability principles are first defined. The usability test is then carried out with each of the three applications under these conditions. The results are then analysed with descriptive statistics. The mean, standard deviation (SD), minimum and maximum values of each application are determined for all principle. Using these values makes a comparison of the three applications possible. Another hypothesis is that the more popular the platform is, the more user-friendly it is. So whether the results of the applications in this usability test correspond to their popularity is also investigated..

In addition to the usability test measures, examining the responses and behaviour of the tested users can provide explanations for the results obtained. This way possible improvements in the usability are identified.

1.5 Structure of the Thesis

In the second chapter the different aspects of usability testing are described. After that the set of usability principles used in the example usability test is defined. More information is given on how this set of usability principles is created. The three music streaming applications are subsequently presented, as well as the research questions. Next, the third chapter presents the methodology. Here, the process of data collection is described, the assessment grid is presented, and the assumptions of the test to be conducted are reviewed. The fourth chapter presents the results. It also discusses them and the advantages and disadvantages of the method used. In the fifth chapter the results are interpreted and recommendations for possible improvements for usability for the three music streaming applications are listed. Finally, the limitations of this usability test and this study are noted. With that come further possible research topics.

Chapter 2

Basic Concepts

In the first section of this chapter usability testing is explained in more detail. In particular the aspects that need to be considered when creating a usability test will be presented. In the second section, the choices for the music streaming applications are presented. The conditions and measurements for the usability test example are defined in the third section. The fourth and final section of this chapter describes the research questions of this usability test example and this thesis.

2.1 Usability Testing

Usability testing includes various methods to test usability design principles. Each method has their certain advantages and disadvantages [6]. When creating a usability test it is therefore important to keep these advantages and disadvantages in mind.

One of the first decisions when creating a usability test is the choice between an expert-based heuristic evaluation or a user-based empirical evaluation. While an expert-based heuristic approach is more resource-efficient, it is not able to achieve the same results as the user-based empirical approach. The two approaches will oftentimes achieve equally good results, but one cannot reproduce the results of the other according to Karat et al. [7]. The experts bring knowledge that the users do not have, and the users bring an authenticity that the experts cannot recreate. The two complement each other when testing different parts of a usability design. The resource-efficiency of the expert-based heuristic approach results from the smaller amount of people being tested. It requires few experts to test a usability principle [8], while it takes several users to test that same principle.

The accuracy is the percentage of problems found out of all existing problems. The optimal amount of users to test in a user-based empirical evaluation depends on the wanted accuracy. The number of usability problems that can be found with n users is given by

$$N(1 - (1 - L)^n)$$

where N is the number of total usability problems of the interface (100%) and L is the number of problems a single user can find out of the remaining problems [9]. The value of L has been empirically determined to be 30% by Jakob Nielsen. According to Jakob Nielsen, the first user tested finds 30% of the problems. However, the second user only finds 21% of the problems. The third finds 15% and the fourth only 10%. The rapid decrease in new detected problems as the number of users tested increases is due to the fact that the users often find the same problems. This means that the accuracy of the usability test can be predicted by the number of users tested. Therefore, the optimal amount of users to test in a user-based empirical evaluation depends on the wanted accuracy. [10]

Now that we have a formula to determine the accuracy of a usability test with the number of users tested, the next question a company could ask is the cost of testing the number of users. The costs for the usability test increase with each user, as the users have to be searched for according to selected criteria and guided through the test. This means the accuracy and cost are proportional to each other. A cost-benefit analysis is therefore sensible prior to finding the users to test with. The cost includes not only the budget but also the time it requires to conduct the test with that number of users. The benefit would be the accuracy of the test which leads to a better usability of the interface. It is precisely this benefit that is one reason why user-based empirical evaluations are often inadequately carried out or neglected altogether, because better user-friendliness does not directly lead to more revenue and the cost of recruiting users is quite high. This is also the reason why heuristic evaluation by experts is often favoured, as it does not require as many test subjects to test the interface. This means that the cost for expert-based evaluations is less than for user-based ones. This can lead to more accuracy with the expert-based methods with equal cost for user-based methods. However, as already mentioned, it does not deliver the same results as a user-based evaluation. The purpose and benefits of user-based methods should not be neglected. [11]

Apart from choosing the number of users, there is also the question of the type of user. Nearly every computer system has a main demographic. This demographic can be determined by various human characteristics, such as their age or gender. Other context-dependent characteristics such as location and time of use of the interface are equally important to consider. Also the reason why they are interacting with an interface is important to keep in mind. The importance comes from the fact that usability can vary between different kinds of users. For instance elderly people are often less experienced with computer systems in general. Additionally the ability to read small text oftentimes decreases with age. The design of an interface that is to be used primarily by elderly people must therefore take their lack of experience and abilities into account. [12]

Another condition that must be defined in addition to the number of user and their characteristics is the way in which the test users are grouped. In the literature there are two designs for conducting an empirical evaluation. The first design is the "between group" design. In this design different interfaces are tested from different groups. Each user tests only one interface. The second design is the "within group" design. In this design each user tests multiple interfaces. Both designs have some advantages and disadvantages. One advantage of the between group design is that there are no distortions due to learning effects. The users each test one interface, while in the within group design they could learn from the previously tested interfaces. This leads to distorted results because they differ from results without such experience. One way of counteracting this is to vary the order in which users test the interfaces. Another benefit from users testing only one interface is that it leads to less fatigue and frustration, which also distort the results. On the other hand fewer users are needed for the within group design, as one user test several interfaces. As we have seen, this is an important criterion in the cost-benefit analysis. Another advantage of a within group design is that it prevents distortion through different skills and knowledge of the users. When a skillful and knowledgeable user only tests one interface, the other interfaces are disadvantaged. Two users never have equal skill and knowledge. That way there will always be a distortion. [13]

The last consideration for the conditions of the framework of the test is where the test is conducted. There are two options. There are lab and field studies. A study where the users come in a set setting to conduct the test is called a lab study. These can be stressful for the users as it sets them in a unfamiliar scenario. In the field study, on the other hand, tests are conducted with the users in their usual environment. This way no additional stress or uncommon factors influence the results. But a disadvantage of this approach is the unequal environment in which the users find themselves during the test.

Now that the evaluation approach, the number and type of users and their grouping have been determined, the conditions for the actual test must be defined. So far, we have been determining the framework conditions. When choosing the user-based empirical evaluation approach the variables that are being analysed have to be defined. Some variables represent different conditions and others are measured in those conditions. The variables representing different conditions are the independent variables. The goal is to find which condition has the best usability. Dependent variables are needed to find this out. These are variables that are measured and analysed. For example, time is a dependent variable that can be a measure of the efficiency of an interface. So for each condition the dependent variables are measured and this results in a difference between the conditions. These results can then be compared and analysed with each other. This means the results from the measured dependent variables tell us what the optimal independent variables are. This also means that you can then make statements about what the best among the tested conditions are. That way the usability for the tested principle is increased using empirical data. [14]

To define the dependent and independent variables we first have to define the usability design principles that we want to test. Many sets of design principles already exist. As mentioned, however, these cannot always be transferred to every interface. Therefore the best way to define a set of principles is to take one or more existing reputable sets and include and exclude principles to fit an interface. To further validate the principles, a comparison between several reputable sets of principles can be helpful. There are not many things to consider when finally defining the set of principles, as all of them lead to a better usability and should therefore be regarded as important. Although there can be a focus on specific principles as it can further benefit a certain user demographic. As already shown an example on how to test a principle would be a time measurement as an indicator for the efficiency of the interface. The aim is therefore to develop a test with which these indicators can be measured efficiently.

Some users will experience problems when interacting with an interface. The problems can be categorised into two possible explanations for the occurrence according to Normans Interaction Model. There are the gulf of execution and the gulf of evaluation [15]. The gulf of execution is the gap between a user's goal and the series of actions they need to take to achieve that goal using the system. The gulf of evaluation is the gap between the system's output and the user's understanding of the output. If the user finds it difficult to bridge this gap, or is unable to do so, this is considered a problem and a flaw in the usability design of the computer system.

2.2 Music Streaming Apps

The usability test example of this paper will be conducted on the following music streaming applications: Spotify, Apple Music and Deezer. The main aim is to test and compare Spotify and Apple Music, while Deezer is intended to be a control group. As we have seen in the previous section, some information about the tested

Music Streaming App	global Monthly Subscribers [in millions]
Spotify	172
Apple Music	78
Deezer	9.7

Table 2.1: global monthly subscribers in the third quarter of 2021

interfaces, in this case the applications, and its users is needed. This section is dedicated to showing important information about the three applications and its users.

All three applications have been in use around the world for years, which means that they are all already well established. Which in turn means that they probably already have good usability, as they have all been constantly improving over the years. One difference between them is that Apple Music is dependent on Apple, whose focus is not primarily on music platforms, while Spotify and Deezer are independent music streaming platforms.

Even though all three music applications are used globally, there is a large difference in popularity between the applications. In Table 2.1 one can see the difference in popularity between the applications, which are shown with the number of global monthly subscribers as of the third quarter of 2021. One can see that Spotify being the largest music streaming platform has had 172 million global monthly subscribers in that time [16]. The second largest music streaming platform in the world Apple Music has only had 78 million global monthly subscribers at that time [17]. This means Spotify was more than twice as popular globally at that time. As Deezer is the control group in this test, it should be an interface that neither Spotify nor Apple Music users are familiar with. This is why Deezer with 9.7 million global monthly subscribers at that time was chosen, as a smaller and lesser known platform in comparison to the other two [18].

The popularity of the three music streaming platforms was represented by the number of global monthly subscribers. However, defining popularity is inherently subjective. Alternative metrics can be considered. For instance, global monthly listeners include users who have not paid for a subscription. Another approach would be a business perspective on popularity, where profit is the major factor. This approach acknowledges that subscriptions on different platforms may vary in cost. However, in this paper, popularity is defined by the count of global monthly subscribers. This choice aims to show a hierarchical comparison. The other metrics mentioned as potential measures of popularity result in a similar hierarchy. Hence, this paper focuses on a singular measurement for the representation of popularity.

Now knowing the popularity of the three music streaming platforms, it is important to mention the launch date of the three applications. Deezer was the first one of the three to launch, starting in 2007. Spotify followed in 2008, one year after Deezer. Apple Music, however, did not launch until the year 2015. This means that Apple Music reached its popularity in approximately half the time of Spotify and Deezer. The main reason for this is that Apple already had a presence in the app store market so pushing Apple Music was easier.

The next step is to discuss the users of these applications. In the previous section, we saw that it is important to identify a main user demographic before starting a usability test. While it is hard to identify a main user demographic between three different applications, there are some characteristics to users of music streaming in general. The first thing is that music streaming applications are most popular to users in between the ages of 18 to 34. The people of this age range make up 55% of the total users of music streaming services [19]. It is hard to find other characteristics shared between many users, since people form all different backgrounds listen to music. For instance there is no gender where music streaming services are more popular. The amount of usage per day also varies heavily.

2.3 Research Questions

This thesis aims to examine how a usability test could be created and conducted. This thesis can serve as a guideline for creating and conducting a usability test. With that it should also help address the problem of neglect as mentioned in the introduction. To this end, it is equally as important to understand the tested interface as its users. Only then can a usability test be efficient and meaningful, as it better represents the actual behaviour of the users with the interface. All this is shown with an example of the music streaming applications Spotify, Apple Music and Deezer. Therefore, in this thesis, the usability of different music streaming applications and their main user demographic is examined in more detail. For this purpose, the following three questions were derived:

Research Question 1: How is a possible usability test for music streaming applications created and conducted?

Research Question 2: Which of the three music streaming applications Spotify, Apple Music and Deezer performs best in the usability test example?

Research Question 3: Do the results correspond to the order of popularity of the three music streaming applications?

In order to create a basis for the study, the first research question deals with the process of creating and conducting a usability test for music streaming applications. Understanding the steps involved in creating a usability test is crucial as it forms the basis for subsequent evaluations. Building on the findings from the first question, the second research question aims to evaluate the performance of three prominent music streaming applications, namely Spotify, Apple Music, and Deezer, through the usability test conducted. Evaluating the usability of these applications is an essential part of gaining practical insights into the strengths and weaknesses of their usability. In order to explore the wider impact of usability in real-life scenarios, the third research question investigates whether the results of the usability test match the popularity ranking of the music streaming applications. Understanding the relationship between usability and popularity, contributes to discussions about user preferences and the success of these applications in the marketplace.

Chapter 3

Novel Method

In this chapter the usability test example is developed. It describes how the test is composed and how it is conducted. This is done so the first research question can be answered. It also discusses how the results will be gathered and how they will be structured. To do this the first section follows the steps discussed in section 2.1 step by step. This sets the conditions for the usability test example. The second and third section present two sections of the test: First the tasks that have to be completed during the test are listed. After the tasks comes an interview where the users are questioned regarding their experience whilst executing the tasks. This interview and its questions are discussed in the third section.

3.1 Creating a novel method for testing usability

In this section the various aspects of usability testing discussed in section 2.1 are followed with the intent to create a usability test for the three music streaming applications Spotify, Apple Music and Deezer.

The first decision was the choice between an expert-based heuristic evaluation or a user-based empirical evaluation. As this paper is intended to serve as an example of an empirical evaluation, this decision has already been made.

After deciding on the evaluation approach, the next question is to determine how many users should be tested. The usability of the three applications will be thoroughly tested to possibly help other developers create their own test. It could serve as a guide for other usability tests, with similar desired accuracy. However, the test could still serve as a useful guide for usability tests with less desired accuracy, as the accuracy can easily be reduced by decreasing the number of users tested.

Following things must be taken into consideration to get a suitable number of users to test. As the test includes various applications, users are divided into groups depending on which application they use in everyday life. To avoid favouring one of the application over the other, the amount of users who use Spotify in their daily lives is the same as the users who use Apple Music. Deezer is not taken into account because it serves as a control group. The number of users must therefore be even. When inserting even numbers of users into Nielsen's Formula, the number 12 is the first to reach an accuracy of about 99%. As a reminder, this is the percentage of usability problems found out of all existing problems.

In addition to accuracy, there are two arguments in favour of the selected number of users. One argument is the cost after twelve users. Each additional tested user adds a negligible accuracy and is not worth the effort. The cost for scope of this thesis would increase too much. In the context of a bachelor thesis the cost of this example is solely time. There is not much more time for further user tests. The second argument in favour of the selected number of users is the fact that these subjects need to be found, which is a long process. It can be difficult to find users that belong to the main demographic.

The type of users to test the interfaces with is the next aspect that needs to be considered. As mentioned in section 2.2 the only characterisation that can be made about users of music streaming services is their age. The age range of the main user demographic of music streaming services is 18 to 34. Users of this age range make up 55% of all users. That is why the age of the users who took part in this test are all in this age range. There are not only characteristics shared in the main demographic that need to be considered. If a characteristic cannot be narrowed down, then this is just as much a characteristic of the main demographic. To respect that the following things are also considered. The first point is gender diversity. As already mentioned in section 2.2, music streaming services are not more popular with any gender. That is why the genders of the users are divided equally. The amount of usage per day is hard to verify in a fair way between the applications. The only thing considered in this aspect is that there are no edge cases where a user hardly uses the application or uses it exceptionally often.

Based on the characteristics of the users the grouping can be made as a next step. As a reminder two designs of grouping the users were described: The between group design where different interfaces are tested by different users and the within group design where multiple interfaces are tested by the same users. To decide between the two discussed grouping designs we can look at some characterisations of the users which counteract some of the disadvantages. One of them is the difference in daily usage. As mentioned in the Basic Concept chapter the difference in knowledge and skill can lead to distortions in the results when using a between group design. With the fact that it is hard to find and to define users with similar amount of daily usage this would further increase the distortion of the results. So this is one indicator to use a within group approach. The fact that the users are already grouped into the two applications brings a similar argument for the within group approach. As already mentioned, the cost for finding and testing further suitable users can lead to time problems in the scope of a bachelor thesis. This is another reason why the within group approach is more suitable in this case. To minimize the distortion through the learning effect, the order in which the users are to test the applications is randomly selected. There are six possible orders in which the three applications can be tested. Due to the fact that both the Spotify and Apple Music group each have six people, each order is conducted once per group. This way none of the applications is favoured.

Once the grouping for the empirical evaluation has been determined, the last step is to define the principles to be tested and the variables for their measurement. Before starting to define each principle and their respective variables, the independent variables can be defined for all principles. In this example, the independent variables are simply the different applications, since they are to be compared to each other. In order to define the principles three existing principle sets are taken and compared with each other to create a new one. The three sets are the "10 Usability Heuristics for User Interface Design" by Jakob Nielsen [20], the "Eight Golden Rules of Interface Design" by Ben Schneiderman [21] and the set from the International Organization for Standardization [22]. To carry out a comparison between these three sets, a comparison was first made between each pair to create three new groups. Two of the sets created in this way were then compared and combined. This and the remaining previously created set were finally compared and combined to create the set used in this usability test. That way each principle of each set is sufficiently taken into consideration. As the main demographic of music streaming applications does not prefer specific features, no special considerations are needed when defining the principles.

The test is split into two parts, one where the users are completing a list of tasks on the specific application and one where they answer questions regarding their experience during the tasks. This separation, as we will see, is due to the fact that not all variables can simply be measured from the physical part where the user completes the tasks. The measurement in the interview section consists of ratings, as users have to rate statements on a scale of one to four. That way they can empirically be compared afterwards.

In addition a second run of the test is carried out. The second run is carried out about four weeks after the first run. This allows for further comparison and more detailed analysis. Now follows a list of the set containing eight principles:

1. **Controllability**
The controllability principle means that users control the computer system and not the other way round. This is particularly important for experienced users. They want to feel as if they mastered the application and that it reacts to their actions. The most important thing to achieve such a feeling is the ability to end the current interaction at any time or the ability to easily undo an action that has been carried out. These things in turn are highly preferred by new users as they encourage exploration. Since this principle describes a feeling rather than a physical property, it is best to measure it in a non-physical way. That is why this is measured in the interview part.

2. **Consistency**
The consistency principle describes the importance of a uniformity. Many things are needed to be consistent. Consistency on in graphical things like design or layout is not enough. The sequence of actions must remain similar. The terminology in the system should also always have the same meaning so that the user can rely on it. These are all characteristics of internal consistency. There is also external consistency, which is just as important, if not more so. According to Jakob's Law [23] a user spends most of their time on other computer systems. This means that users prefer computer systems to work the same way as the others they already know. For the same reason as controllability, this principle is also measured in the interview.

3. **Visibility**
The visibility says that the status of a computer system should always be visible to the user. This also requires a visible update with every user action, because the status of the computer system changes with every action. So for every action of the user there should be a feedback from the computer system. The size of this feedback should be proportional to the size and seriousness of the action of the user. Feedback should also be provided quickly. According to Robert B. Miller the longer a feedback takes, the more distracting it is to the user's train of thought [24]. The same argumentation applies to the measurement as to the two previous principles for visibility. Visibility is therefore also measured in the interview.

4. **Unencumberedness**
The unencumberedness principle is a combination of many aspects from other principles. This principle states that the use of a computer system should never be mentally or physically demanding. To achieve this, a computer system should especially take the following two points into account. The first is that a system should

not be too heavy for a user's memory load. It is less tiring for a user to recognise something than to remember it. To do that the internal and external consistency are important as well as the visibility. The second point is that it is important to let the user know when one task is finished and they can move on to the next one. This way, if they move on to another task, they do not have to remember what the task was, but can write the task off as completed and devote themselves entirely to a new task. This principle is the last one that is measured with the interview part.

5. **Efficiency**
Every computer system has a purpose and the user interacts with it to satisfy a certain need. The user wants to satisfy that need as fast as possible. This is why a computer system should be efficient. The efficiency principle describes that user needs should be satisfied quickly and without detours. This should be the case for all types of users. This means that, more efficient ways to execute a task should be available to also satisfy more experienced users. The measurement of this principle is not part of the interview. This principle is measured purely by the time it takes for a user to complete all the tasks. This way both the time of experienced users, who are already using the application, and that of new users is taken into account. It is important to mention that the time spent reading the tasks is also counted.

6. **Learnability**
The learnability principle is similar to the efficiency principle. However, this principle is focused on new users. A computer system should be designed in such a way that it can be used intuitively. It should encourage learning through exploration instead of forcing a way how the user must learn. This way different types of users can interact with the computer interface at their own pace. The measurement for this principle is also the time it takes for the users to complete the tasks. However, only the time for the new users is taken into account. An additional crucial aspect of this principle is the ease with which information previously learned from the computer system can be memorized. This attribute was assessed during the second run of the test. The enhancements observed from the first to the second run can serve as indicators of memorized information from the initial session.

7. **Error Tolerance**
The error tolerance principle consists of two major points: Error prevention and error correction. If an error happens the user should have minimal time and effort to correct it. The computer system should do as much as possible to correct itself. It would be even better to prevent errors from occurring in the first place. There are

two types of errors users can make: Slips and mistakes. Slips are unconscious and happen when the user has not payed enough attention. These can be prevented by setting helpful defaults or setting certain constraints. The second type are mistakes. These are conscious errors and occur when the mental model of the user and the interface do not match together. This principle is measured by adding up the time that users need to recover from their errors. This way the amount of errors and the time it takes to recover are both taken into consideration. In this way, many small errors and a single large error are weighted equally. This is not always the case and is a simplification due to the nature of the applications. Computer systems where a major error would lead to more stress for the user would have to be weighted differently.

8. **Flexibility**
The flexibility principle means that users should have multiple ways to complete the same task. This way each user can pick the method that suits them the best. This is also a way to make it easier for expert users to speed up the whole process. An example for this are shortcuts. This was measured by the different paths users took to complete the tasks. All the different paths are added together to obtain a number as a result. But also a path tree is meaningful, as it shows which tasks are more flexible in comparison to the others. Another point of flexibility is the ability to modify the computer system to the specific needs of the users. This way the interface can be adapted to each user's preferences. This part was not tested in the usability test example, because it is too difficult to test empirically. This is an aspect where it would be more efficient and accurate to test it heuristically with experts.

As we have now the set of principles for the usability test example, the next thing is the analysis of the results. The results of the test are mostly numerical. That way the mean, standard deviation (SD), minimum and maximum can be defined for each principle. These are compared and analysed via descriptive statistics. More complex statistics are not suitable due to the small amount of data. With this small data set, the accuracy and reliability would be insufficient.

3.2 Task Completion Section of the Test

With all the circumstances and measure methods defined the next thing are the specifics of the test and its process. This is why the next two sections will discuss the task list for the users to complete and the questions during the interview.

To test the three applications Spotify, Apple Music and Deezer, users completed a list of tasks in each application. These were simple tasks, some of which could be completed with a single action. During the design of the list of tasks a conscience effort was made to involve different aspects and menus of the applications. This way more possible problems could be found. All three applications were tested on an Android mobile phone that was provided to the subjects. None of the users tested use android in their everyday lives. This way, all users have the same starting conditions during the test. In addition, the applications were not updated during the entire usability test, so the same versions of the applications were used for all tests. The test was conducted in a space with no distraction to further reduce differences in conditions. This means it was a lab study as opposed to a field study. The screen of the phone was recorded during the completion of the tasks, allowing for accurate time measurement. In addition, this approach allows for the analysis of problems encountered by users. The users were also observed while completing the tasks. Through their behaviour and reactions, problems in crossing the gulfs of execution and gulfs of evaluation were found. These can additionally be analysed alongside the recording of the screen. These are then described as potential usability improvements in respective application.

Now that the sequence of the task part is presented, the next thing to discuss are the tasks itself. Table 3.1 shows all 14 tasks that a user are told to complete in the Spotify application. These had to be done in numerical order. Some of them are simple, like task number 13, which tells the users to play the most popular song on the page of an artist. Task number 3 on the other hand, tells the users to create a new playlist, name it a certain way and add a song to the playlist. This contains several steps and allows the users to take different approaches to complete it. The task lists for the other two applications contain similar tasks. However, the tasks are always slightly modified and worded a little differently. This counteracts the learning effect somewhat, as the users still have to read the tasks thoroughly.

3.3 Interview Part of the Test

After completing the task list for an app, an interview is conducted where the experience during the completion of the task list part is discussed. The interview section is followed by the same procedure for the next application. The interview section consists of statements that had to be rated on a scale of one to four, with one being not true and four being completely true. Users had the opportunity to change their ratings and compare them with the ratings they gave to the other two apps. This way, their ratings were not fixed until the end of the entire test. The

Spotify	
1.	Log in with the following login details: Email: bscusability@gmail.com Password: Ab12Cd34.
2.	Play the song *“They Don’t Care About Us”* by Michael Jackson.
3.	Create a new playlist. Name it *“Michael”* and add the song from Task 2.
4.	Add all the songs from Michael Jackson’s album “Thriller” to the playlist.
5.	You change your mind and decide not to include the song *“Wanna Be Startin’ Somethin’”* in the playlist. Remove it from the playlist.
6.	Change the audio quality in the settings to a medium audio quality. Do this for all types of internet connections.
7.	Search for the playlist with Hip-Hop songs from the 90s created by Spotify.
8.	Play the playlist in shuffle mode.
9.	Skip three songs.
10.	Go back two songs and add it to your favorites.
11.	Find the artist of this song and play another song by them.
12.	Search on the artist’s page for suggestions for similar artists. Choose one of them and go to their profile page.
13.	Play their most popular song.
14.	Add the next two most popular songs to the queue

Table 3.1: List of tasks that had to be completed in the Spotify application during the usability test

ability to compare their ratings led to more accuracy, as some users find it difficult to rate something without comparison. The previously discussed gulfs were another help for the users to rate the statements. If there was a problem crossing one of the gulfs during the completion of the task, the gulfs were used as a reason for giving a worse rating.

Knowing the circumstances of the interview part, the statements from the interview are the next thing to discuss. Table 3.2 lists all of the statements from the interview. There are always three or four statements for each principle. These represent characteristics of given principle. The sum of the scores for the statements is the metric for the principle to which the statements belong. Unlike the task list, the statement in the interview remains the same for all applications. A change in interview questions between platforms would not benefit any metric, but would only lead to confusion and remove the possibility of comparison. The statements are grouped together according to the respective usability principle. Doing so gives context of the respective usability principle which helps some users understand the statement better. The categories also summarise the statements that are used to measure the respective principle.

Controllability	
1.	There were no restrictions from the application, and I could always perform what I wanted freely.
2.	I felt as if I always had control. I could always decide what the application should do, not the other way around.
3.	I had no fear of performing actions because I was aware that actions were easily reversible.
Consistency	
4.	Experiences I had outside of this application helped me solve the tasks.
5.	I felt that the application was consistent between different menus.
6.	The unity between menus aided me in orientation and usage.
7.	I could always predict what would happen and how the application would respond to an action.
Visibility	
8.	I always felt like I understood what was happening or what the application was doing.
9.	I felt that the application helped me understand what was happening and what the application was doing.
10.	After every important action, the application informed me of what happened and what the application was currently doing.
11.	I never waited for a response or feedback from the application.
Unencumberedness	
12.	I was aware when a task was completed, allowing me to move on to the next one.
13.	There were no moments during the use of the application that I found mentally/physically demanding.
14.	The use of the application required no concentration from me.

Table 3.2: List of statements during the interview of the usability test

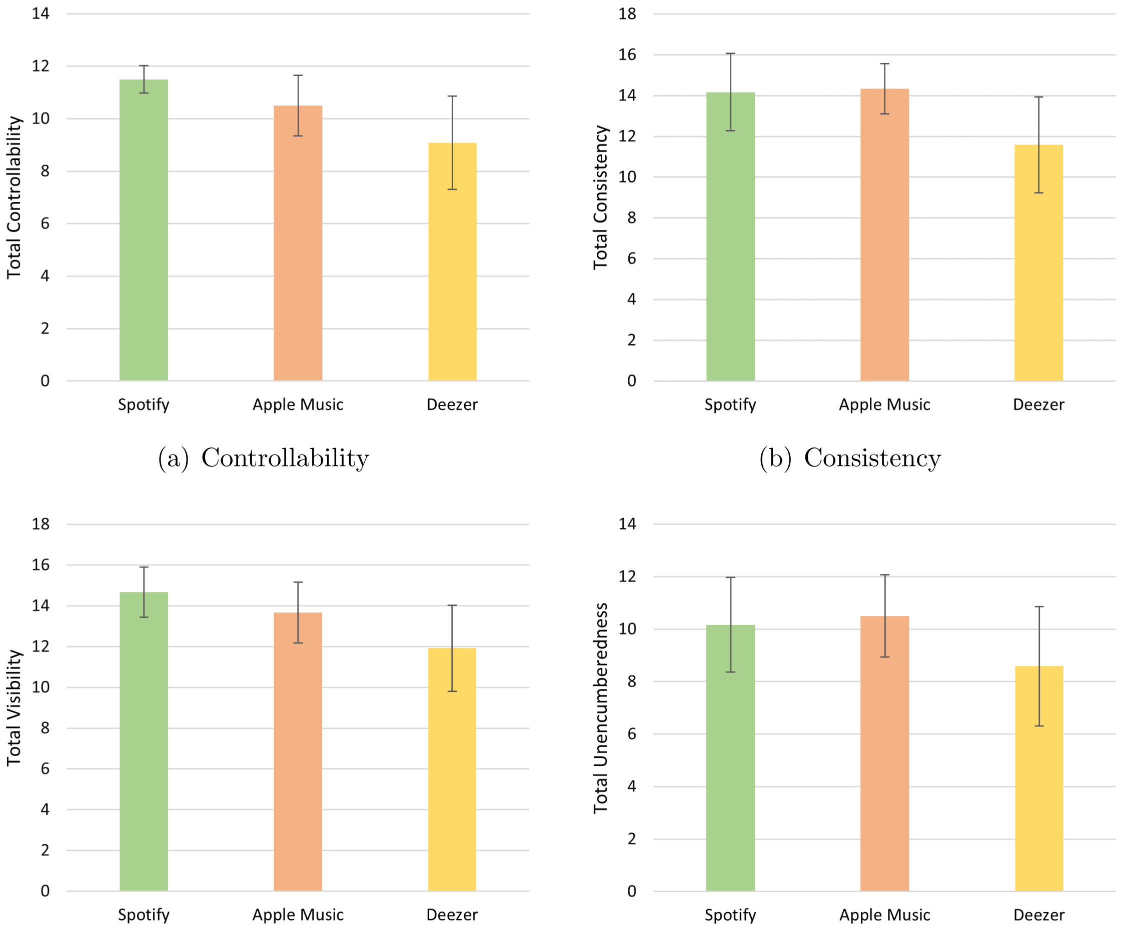
Chapter 4

Experimental Evaluation

This chapter will illustrate and discuss the results of the usability test to answer the second and third research question. First the results of the eight tested principles are each presented with a figure. These figures are described to answer the second research question of which applications performs better in the usability test. Seven of the figures are bar charts. Each bar chart contains three different coloured bars representing the three tested applications. The green bar represents the results from Spotify, the red bar represents the results from Apple Music and the yellow bar represents the results from Deezer. All the bars represent the mean of the results of all users. There is an error bar on each of the three bars that represents the SD. In addition to describing the results, it also explains why the results turned out the way they did. This is done using several examples.

The first four figures show the data on the first four principles. The first four principles are those that were tested in the interview section. As a reminder, these are ratings of certain features that constitute good usability. This means that a higher score indicates better usability.

Figure 4.1 (a) shows the mean total scores of the controllability principle. The overall mean score for Spotify is 11.5 with a SD of 0.522, a minimum of 11 and a maximum of 12. The overall mean score for Apple Music is 10.5 with a SD of 1.16, a minimum of 8 and a maximum of 12. The overall mean score for Deezer is 9.08 with a SD of 1.78, a minimum of 6 and a maximum of 12. Spotify has the highest mean total result and the least variability. This indicates that Spotify has the best controllability among these three applications. Apple Music follows Spotify with a slightly lower mean but a larger variance. Deezer is in last place. Deezer received a lower rating due to its behavior in certain menus where it forcibly scrolled the screen to a specific position. In these instances, when users attempted to scroll and released, the screen automatically moved to the predetermined position, resulting in a loss of user control. This led to frustration and a feeling of restriction for some



(a) Controllability

(b) Consistency

(c) Visibility

(d) Unencumberedness

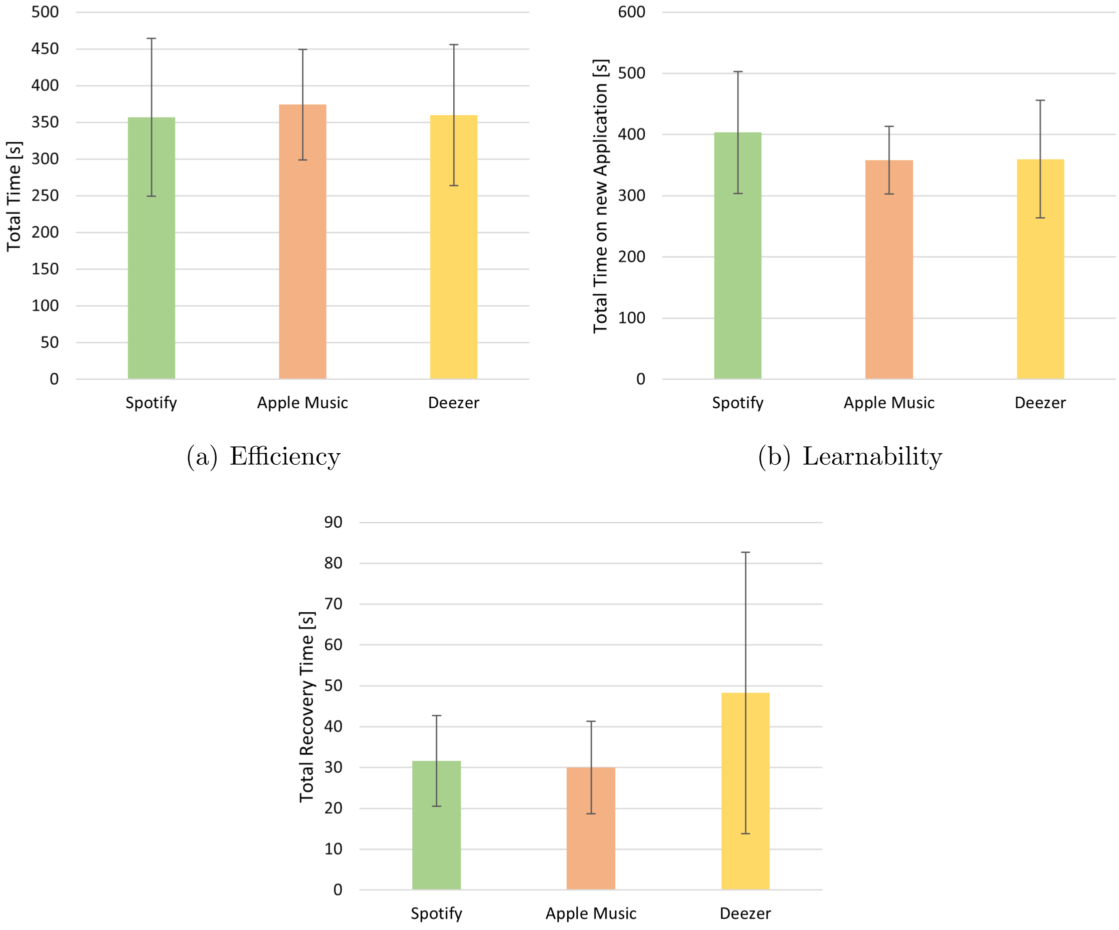
Figure 4.1: Main caption for the entire figure

users. Users who encountered this issue expressed their frustration either at the time of occurrence or later during the interview. On Apple Music some new users had problems creating a playlist for Task 3. The playlist had to be confirmed in a way that some users did not understand, which also led to a feeling of restriction, as some users stated in the interview.

Figure 4.1 (b) presents the overall mean of the ratings of the consistency principle. Spotify has a overall mean score of 14.17 and a SD of 1.9. The minimum for spotify is 10 and the maximum is 16. For Apple Music the overall mean score is 14.33 with a SD of 1.23. The scores range from a minimum of 12 to a maximum of 16. This suggests that Apple Music is slightly more consistent than Spotify, as the overall mean is higher and the variance is lower. Deezer scores the lowest with a overall mean score of 11.58 with a SD of 2.35, a minimum of 8 and a maximum of 15. Spotify and Apple Music had only minor complaints, hence the similar result. Deezer had one major problem in external consistency. The menu where all saved and liked music was found is called favorites and has a heart icon whereas in Spotify and Apple Music it is called library.

The next Figure, Figure 4.1 (c) shows the overall mean scores for the visibility principle. The overall mean score for Spotify is 14.67 with a SD of 1.23. It scored a minimum of 13 and a maximum of 16. For Apple Music the overall mean is a bit lower with 13.67. It has an SD of 1.5 and a minimum of 11 and a maximum of 16. Deezer again finishes last with an overall mean score of 11.92, an SD of 2.11, a minimum of 10 and a maximum of 16. These values indicate that Spotify has the best visibility, followed by Apple Music and then Deezer. The same problem as in controllability for the playlist creation on Apple Music is the reason for the lesser rating in this principle. Some users were not sure if they were done creating the playlist or not. Additionally the small feedback after every action was better perceived on Spotify. With Deezer, the main problem was that some users were waiting for no reason because in some cases the application did not inform them what was happening.

On Figure 4.1 (d) the overall mean scores for the unencumberedness principle is presented. The Spotify results give a mean score of 10.17 and a SD of 1.8. The minimum total rating is 6 and the maximum is 12. Apple Music is a bit better with mean and SD of 10.5 and 1.57. Deezer is behind with a overall mean of 8.58 and a SD of 2.27. The minimum is 4 and and the maximum 12. This suggests that Apple Music fulfills the unencumberedness principle the best. A bit behind is Spotify and then comes Deezer. As described in the definition of this principle, it is a mix of many aspects of other principles. One thing that had a big impact on this was task number 6, where users had to change the quality of the sound. The settings on



(a) Efficiency

(b) Learnability

(c) Error Tolerance

Figure 4.2: Main caption for the entire figure

Spotify are all listed together and thus form a long list. The audio quality settings are almost at the bottom of this list. For this reason, many users had to read a lot and concentrate to find them. Hence the lower rating. With Deezer, there was no main problem like with Spotify. Since the unencumberedness principle combines various aspects of other principles, issues in these aspects contributed to Deezer's overall rating.

Now that we have seen the four principles tested in the interview, here are the other four that were tested with the task list. These were all measured with time except for the flexibility principle. As mentioned in the definition of the principles, less time is preferable. For this reason, a lower mean value is considered better in the following three figures.

In Figure 4.2 (a) the mean of the time it took the users to complete the tasks can be seen. This represents the efficiency principle. The users took an average of 357 seconds to complete the task list for Spotify. The SD value is 107, the minimum is 208 and the maximum is 572 seconds. To complete the task list for Apple Music the users took 374 seconds on average. Here the SD value is 75, the minimum is 262 and the maximum is 529 seconds. This indicates that Spotify is a bit more efficient than Apple Music. The average duration it took the users to complete the tasks for Deezer is 360 with a SD of 96. The minimum for Deezer is 234 seconds and the maximum 612 seconds. This puts the average duration on Deezer in between the other two in the ranking. The results to this principle are the most balanced. There was no big design problem that lead to a big difference in one of the three applications.

The learnability principle has the same measurement method as the efficiency principle. However, only the measures of users who do not use the respective app on a daily basis are taken into account. Figure 4.2 (b) shows the mean of the time that new users took to complete the tasks in the applications. For Spotify it took new users 403 seconds to complete the tasks. The SD is 100, the minimum time is 257 seconds and the maximum is 572 seconds. For Apple Music the mean is 358 seconds with a SD of 55, a minimum of 262 seconds and a maximum of 419 seconds. This puts Apple Music at first place with respect to the learnability principle. On Deezer all the users were new users, meaning the results remain the same as in the efficiency principle. With these results Deezer is second and Spotify at last. As this is the same measure as the efficiency principle there also were no problems here worth mentioning. However, it's worth noting that Spotify ranks last on this principle, but first on the efficiency principle. This means that it is less friendly for new users, but experienced users get much better results, so the average completion time across all users is lower than the others.

Figure 4.2 (c) represents the seventh principle, the error tolerance. The bars present the mean of the total time it took the users to recover from their errors. On Spotify it took the users 32 seconds on average to recover from their errors. This has a SD of 11, a minimum of 14 seconds and a maximum of 51. Apple Music has similar results with the average being 30 and a SD of 11. The minimum on Apple Music was 9 seconds and the maximum 45 seconds. So Apple Music is a little bit better when it comes to error tolerance. Deezer comes in last with a larger difference. The mean time for error recovery on Deezer is 48 with a SD of 34. The minimum amount of time was 24 seconds and the maximum was 149 seconds. This large difference between Deezer and the other two is best represented with the maxima. The maximum for Deezer comes from one user unintentionally deleting the created app instead of removing one song from it in task number 5. The user had to recreate the playlist from start again. This lead to the large time loss.

The results for the last principle, flexibility, are represented in Figure 4.3 with

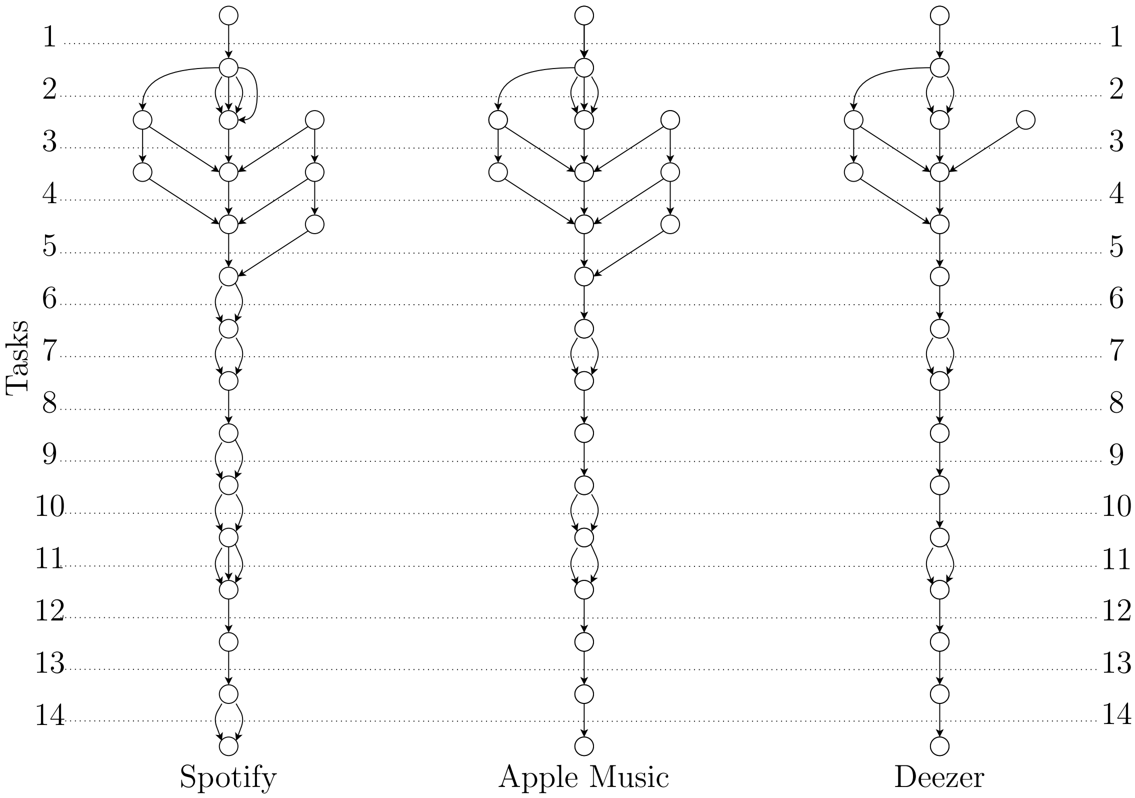


Figure 4.3: Usability Evaluation Results for Principle 8. *Flexibility*

path tree diagrams. They represent all possible ways the users took to complete the tasks. Each level represents a specific task. The three path trees represent the three applications. Each node is a state or a menu of the application. Each edge is a method to complete a task leading to a new state of the application. Counting all edges together gives a total of 33 edges for Spotify, 28 for Apple Music and 22 for Deezer. This shows that Spotify was more flexible for users than the other two, while Deezer was the least flexible and Apple Music was right in the middle. Apart from singular extra methods on some tasks the biggest difference can be seen on the second level on the path trees. This is the second task where the users had to play a specific song. To do this, they had to use the search function. There, users usually searched for the title of the song, the artist or both. On Spotify and Apple Music, users also searched for the artist and looked for the song on the artist's page. On Spotify, one user even searched with the lyrics of the song. Another notable method on Spotify is on task 6, where one user found the settings using the same search function.

A second run of the usability test was conducted. The results from this run had very minor differences and were therefore not listed and discussed here. They were random so they showed no significant results and had no possible explanation. The problem with the second run was the time between the two runs. With four weeks in between the runs the users forgot most of the things they learned from the first run. This lead to similar experiences and results. An improvement could be a smaller delay between the two runs.

Now that all results have been presented and additional explanations were given, the evaluation and its results can be discussed. The biggest limitation that should be mentioned is the fact that only Spotify and Apple Music users took part in this usability test. This leads to distortions that cannot be avoided, except by involving deezer users, but this would be too costly and would go beyond the scope of this thesis. A mistake in the design of this evaluation design was the aforementioned second run of the test to test the learnability. Apart from this, there are no distortions or disadvantages worth mentioning. Almost all results could be justified with the help of certain aspects of the designs of the applications. For this reason, the evaluation went as planed and fulfilled its purpose of comparing the applications and finding possible design improvements.

Chapter 5

Conclusions and Future Work

This chapter discusses the findings and implications of this thesis. Furthermore, the limitations are pointed out and questioned critically, followed by recommendations for further research.

5.1 Findings

In this thesis the creation and conduction of a usability test on music streaming applications was examined. Various possible approaches and considerations were analyzed to create the test. Here, the first research question regarding the creation and conduction of a usability test for music streaming applications comes up. The usability test was created by systematically following existing approaches and considerations, supplemented with additional information about the tested interfaces. This way, all the conditions for the usability test were set. The measurements and measurement methods for the dependent variables were determined according to the characteristics of each principle. This led to the division of the usability test into two parts: a task-oriented part, where users completed a list of tasks, and an interview part, where users provided ratings based on their experience during the task completion. The test was then conducted with a total of twelve users, carefully selected to fit all the previously set criteria.

This leads to the results of the conducted usability test, providing an answer to the second research question about the performance of Spotify, Apple Music, or Deezer in the usability test. To answer the question the results were first analysed with descriptive statistics. For each principle the mean of the results was described and then compared. The results were explained using specific design aspects of the applications where possible. This way possible improvements in the design of the interfaces were found. The comparison between the three happened individually for each principle and could be quantitatively rated. Spotify and Apple Music each have the best results in four of the eight principles, making them equally user-friendly. While there was not a substantial difference in their results across any of the principles, the contrast between Spotify and Apple Music and the third tested application, Deezer, was notably larger. Therefore, the answer to the second research question indicates a tie between Spotify and Apple Music, while Deezer performed a bit worse.

With the answer to the second research question comes the third and final research question, regarding the correspondence of the results of the usability test to the popularity of the three applications. Deezer, being the least popular globally measured by the number of global monthly subscribers, matches the results of the usability test. The similar results from Spotify and Apple Music in the usability test, however, do not have a correspondence to the difference in global popularity between these two applications.

The usability test has some disadvantages and has improvements in both framework conditions and execution. Other than those, the usability test did bring meaningful results and fulfilled its purpose. Therefore, it can be considered a successful usability test with possible improvements. The main possible improvements that were found with this test are the following: On Spotify, most users had difficulties navigating the settings because it is a long list to navigate through. A hierarchical arrangement of the settings in folders would make it easier for users to find a specific setting. In the Apple Music application the feedback could be somewhat improved by making it stand out more from the rest of the user interface. This would be particularly important for creating and editing playlists, as most users struggled with a feeling of uncertainty while completing these tasks. For Deezer, the proposed improvement is better consistency both internally and externally. For example, the names and icons of the various menus caused confusion for some users. The recommendations in the search function also confused many users as they were unusual for them. This improvement would make it easier for new users in particular to get started with the application and possibly lead to more users sticking with it. This would therefore lead to greater popularity.

5.2 Limitations and Suggestions for Further Research

There is one main limitation in this work, namely the non-inclusion of deezer users. The usability test example was testing the usability of the three applications Spotify, Apple Music and Deezer. However, the test was only carried out with Spotify and

Apple Music users. This leads to some distortion in the results. The distortion arises because certain design principles provide guidelines on how to cater to experienced users. This becomes problematic when the expert users of an interface are not included in the test, resulting in distortions as the aspects of the principles related to expert treatment are overlooked. In this usability test these aspects of the principles were therefore not testable for Deezer. This is mainly the case for the flexibility principle but also influences other principles such as efficiency.

Another limitation is the focus on user-based usability testing. This paper focusses on user-based empirical usability evaluation because it is neglected. However, this does not mean that user-based methods are to be favoured. An extension with expert-based methods would certainly be helpful to identify more improvements.

Apart from that some further research could be done to improve or further validate the elaborated methods of this work. A first possible continuation would be to implement the suggested improvements and reconduct the test. This way the improvements can be checked for their effectiveness. If the results are positive, the effectiveness would further validate the elaborated methods of this work. Another possible further research could be the creation of another usability test to examine the elaborated methods and its functionality. Another possible research idea is to use the usability test created in this thesis on other platforms, like video streaming sites. This could help us understand how well the usability principles apply to different types of websites.

To summarise, the example and methods for user-based empirical usability evaluations are reasonably effective and lead to helpful results. Since the usability test in this thesis was created and conducted by a single person as part of a bachelor thesis, it may reduce the fear of high costs and motivate others not to neglect this important part of HCI. In addition, the methods used in the usability test example in this thesis can be used as a guideline for the creation and implementation of further usability tests.

Bibliography

[1] Alan Dix, editor. *Human-Computer Interaction.* Pearson/Prentice-Hall, Harlow, England ; New York, 3rd ed edition, 2004.

[2] Marc Hassenzahl and Noam Tractinsky. User experience - a research agenda. *Behav. Inf. Technol.*, 25(2):91–97, 2006.

[3] Jakob Nielsen. *Usability Engineering.* Academic Press, 1993.

[4] Clare-Marie Karat. Chapter 32 - Cost-Justifying Usability Engineering in the Software Life Cycle. In Marting G. Helander, Thomas K. Landauer, and Prasad V. Prabhu, editors, *Handbook of Human-Computer Interaction (Second Edition)*, pages 767–778. North-Holland, Amsterdam, 1997.

[5] James R Lewis. Usability testing. *Handbook of Human Factors and Ergonomics*, pages 1267–1312, 2012.

[6] Christopher Jewell and Franco Salvetti. Towards a combined method of web usability testing: an assessment of the complementary advantages of lab testing, pre-session assignments, and online usability services. In Joseph A. Konstan, Ed H. Chi, and Kristina Höök, editors, *CHI Conference on Human Factors in Computing Systems, CHI '12, Extended Abstracts Volume, Austin, TX, USA, May 5-10, 2012*, pages 1865–1870. ACM, 2012.

[7] Clare-Marie Karat, Robert Campbell, and Tarra Fiegel. Comparison of empirical testing and walkthrough methods in user interface evaluation. In Penny Bauersfeld, John Bennett, and Gene Lynch, editors, *Conference on Human Factors in Computing Systems, CHI 1992, Monterey, CA, USA, May 3-7, 1992, Proceedings*, pages 397–404. ACM, 1992.

[8] Jakob Nielsen. Finding usability problems through heuristic evaluation. In Penny Bauersfeld, John Bennett, and Gene Lynch, editors, *Conference on Human Factors in Computing Systems, CHI 1992, Monterey, CA, USA, May 3-7, 1992, Proceedings*, pages 373–380. ACM, 1992.

[9] Kaspar Riesen. Mensch-maschine schnittstelle. Vorlesungsskript (5.0), 2022. Herbstsemester.

[10] Jakob Nielsen and Thomas K. Landauer. A mathematical model of the finding of usability problems. In Bert Arnold, Gerrit C. van der Veer, and Ted N. White, editors, *Human-Computer Interaction, INTERACT '93, IFIP TC13 International Conference on Human-Computer Interaction, 24-29 April 1993, Amsterdam, The Netherlands, jointly organised with ACM Conference on Human Aspects in Computing Systems CHI'93*, pages 206–213. ACM, 1993.

[11] Jakob Nielsen. Usability inspection methods. In Catherine Plaisant, editor, *Conference on Human Factors in Computing Systems, CHI 1994, Boston, Massachusetts, USA, April 24-28, 1994, Conference Companion*, pages 413–414. ACM, 1994.

[12] Gaby Anne Wildenbos, Linda W. P. Peute, and Monique W. M. Jaspers. Aging barriers influencing mobile health usability for older adults: A literature based framework (MOLD-US). *Int. J. Medical Informatics*, 114:66–75, 2018.

[13] Gary Charness, Uri Gneezy, and Michael A. Kuhn. Experimental methods: Between-subject and within-subject design. *Journal of Economic Behavior & Organization*, 81(1):1–8, 2012.

[14] Sung H. Han, Myung Hwan Yun, Kwang-Jae Kim, and Jiyoung Kwahk. Evaluation of product usability: development and validation of usability dimensions and design elements based on empirical models. *International Journal of Industrial Ergonomics*, 26(4):477–488, 2000.

[15] Donald A Norman. Cognitive engineering. *User centered system design*, 31(61):2, 1986.

[16] Statista. Number of spotify premium subscribers worldwide, 2023. `https://www.statista.com/statistics/244995/number-of-paying-spotify-subscribers/`, Accessed on December 29, 2023.

[17] Statista. Number of apple music subscribers worldwide, 2022. `https://www.statista.com/statistics/244995/number-of-paying-spotify-subscribers/`, Accessed on December 29, 2023.

[18] Jeronimo Folgueira. Deezer revenue growth in q3 2022. *Deezer Press Release*, 2022.

[19] Thuhin Khanna R, Sundararajan S, and Jayashree K. User demographic analysis of music streaming platforms. *Materials Today: Proceedings*, 62:4953–4956, 2022.

[20] Jakob Nielsen. 10 usability heuristics for user interface design, 1994. `https://www.nngroup.com/articles/ten-usability-heuristics/`, Accessed on December 29, 2023.

[21] Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Niklas Elmqvist, and Nicholas Diakopoulos. *Designing the user interface: strategies for effective human-computer interaction.* Pearson, 2016.

[22] International Organization for Standardization. Ergonomics of human-system interaction — Part 110: Interaction principles. ISO Standard ISO 9241-110:2020(en), ISO, 2020. https://www.iso.org/standard/77520.html.

[23] Jon Yablonski. *Laws of UX: Using Psychology to Design Better Products & Services.* O'Reilly Media, Sebastopol, CA, 2018.

[24] Robert B. Miller. Response time in man-computer conversational transactions. In *American Federation of Information Processing Societies: Proceedings of the AFIPS '68 Fall Joint Computer Conference, December 9-11, 1968, San Francisco, California, USA - Part I*, volume 33 of *AFIPS Conference Proceedings*, pages 267–277. AFIPS / ACM / Thomson Book Company, Washington D.C., 1968.

Erklärung

gemäss Art. 30 RSL Phil.-nat.18

Name/Vorname: Bejta Korab

Matrikelnummer: 19-133-305

Studiengang: Computer Science

Bachelor ☑ Master □ Dissertation □

Titel der Arbeit: Creation of a novel usability test and conducting it on music streaming applications

LeiterIn der Arbeit: PD Dr. Kaspar Riesen

Ich erkläre hiermit, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe r des Gesetzes vom 5. September 1996 über die Universität zum Entzug des auf Grund dieser Arbeit verliehenen Titels berechtigt ist.
Für die Zwecke der Begutachtung und der Überprüfung der Einhaltung der Selbständigkeitserklärung bzw. der Reglemente betreffend Plagiate erteile ich der Universität Bern das Recht, die dazu erforderlichen Personendaten zu bearbeiten und Nutzungshandlungen vorzunehmen, insbesondere die schriftliche Arbeit zu vervielfältigen und dauerhaft in einer Datenbank zu speichern sowie diese zur Überprüfung von Arbeiten Dritter zu verwenden oder hierzu zur Verfügung zu stellen.

11.01.2024

Ort/Datum

K. Bejta

Unterschrift