Assignment 3: Design Reflection

Elia Baisch
2.Semester
Matrikel-Nr: 43506

Content

Persona

Definition:

A persona is a fictitious user who reflects characteristics for a specific target group of people.
They are responsible for developing a product that is more targeted and specifically adapted to the selected customer type.
This allows the work to be much closer to the customer, which has a very positive effect on the customer relationship
and ensures greater satisfaction in the area of use.

Personal benefit:

It was possible to put yourself more in the customer's shoes, because you had a kind of person standing in front of you.
On the basis of the artificially presented biography, it was possible to extract behaviors and other characteristics.
With the help of this information it was possible to create a design according to the requirements.
If the person was neat or messy, then you had to arrange everything neatly for a clear structure
or you could creatively create a little disorder.
Are certain color patterns recognizable that might be important for details in the background.
Everyday life or hobbies were also an important input when it comes to the special function of the app,
with which you can possibly address exactly these activities and make something special out of it,
or that you can apply the function to so many of these activities and even help him with it.
Other aspects, such as the use of other apps, provide information about the navigation
and interaction environments in which the user is located,
making it possible to create a familiar environment that enables the user to quickly find his way around the app.

iterative design approach

Planning

Every project must be planned thoroughly at the beginning, with the help of the specific requirements of the users.
However, these requirements must first be collected and analyzed in order to use them in the further process.
Now it is time to continue the development process and research design elements.
This is also the time to start thinking about usability improvements to enhance the user experience.

Analysis and design

Now, after the design phase, it is time to analyze the usage of the product in order to define the business logic
that enables the communication between the product and the end user. Attention is paid to
the exact usage in order to adapt it perfectly to the user and finally to make first drafts.

Implementation

This phase is very clear, after all the research since the beginning of the development process,
it is now the turn of the defined requirements to be put into practice and to make a first prototype
that is adapted to these requirements in the best possible way to be as close as possible to the perfect product.

Testing

Once the first prototype has been made, it must of course be thoroughly tested for all functions,
visual impressions and general use in order to find and eliminate bugs and errors in the design and function.

Evaluation

Now it is up to the customer to check the final product to see if all requirements have been met to the best of their ability.

If something is missing, the whole process must be repeated.

Use on own project

Here, high and low fidelity prototypes find their exact application. It is up to the developer to start with low fidelity prototypes in order to quickly achieve preliminary results, which in the best case are already close to the perfect result.
This means that it is assumed in advance that the processes must be run through several times in order to design the perfect end product. Therefore, low fidelity prototypes are made as often as possible until the perfect end product is as close as possible, and the whole process is repeated just as often.
When you have reached the point where you have a low fidelity prototype that is as close as you can get to the final product, you then use it to create a high fidelity prototype that represents the final product down to the smallest detail.

Native App

most common type of mobile app
Built for specific platforms
BUilt using native programming languages

Advantages:
Very fast
Built to run on specific platform
Interactive and intuitive
Interact with device utilities

Disadvantages:
Single platform
Hard codeing languages
Developer are expensive
Hard to maintain

Hybrid Apps

combination of natvie and web apps
Use HTML / Css / JavaScript
Ran inside container / webview

Advanteages:
Easy to build through HTML / CSS / JavaScript
Developer are cheaper
Single app for all platforms
No browser needed
Access through all App Stores
Access device utilities using an API
Faster to develop than native apps

Disadvantages:
Slower than native apps
Less interactive than native apps
Not as many plugins to interact with devices

Conclusion:

After listing the advantages and disadvantages of the respective app styles, I would rather choose the hybrid app,
since many more points speak for it. Be it the cost or the effort to create and maintain the application.
Especially since you have the possibility to access it on every platform and with the function
you are not necessarily dependent on a single device, but you can also use this function on a laptop at a party or further.
Of course, it is debatable that the native app can be more responsive to the functions of the individual devices,
but this is hardly required by the app, except that you should be able to play sounds.
A point that might still speak for a native app is if you want to use the function of, for example, t
he smartwatch to capture the pulse of the user and thus perfect the modes,
but this is not available in this version and thus the choice of the hybrid app would be a more sensible decision.

Material Design Guidelines

Properties

Material surfaces have consistent, unchanging properties and behaviors in material design,
which are also reflected in the app and provide an orderly structure without causing confusion.

Shadows, Color and Content

The Guidelines are using shadows to highlight the interaction options, which do not occur in my app,
however these will also be specified by colors in the guidelines, which is more likely to find their application in my app.
However, it was not exactly to highlight functions and interaction possibilities,
but rather to create a contrast to make individual elements recognizable.

However, the content is comparable to the guidelines and is not highlighted by a kind of thickness on a new layer.

Physical properties

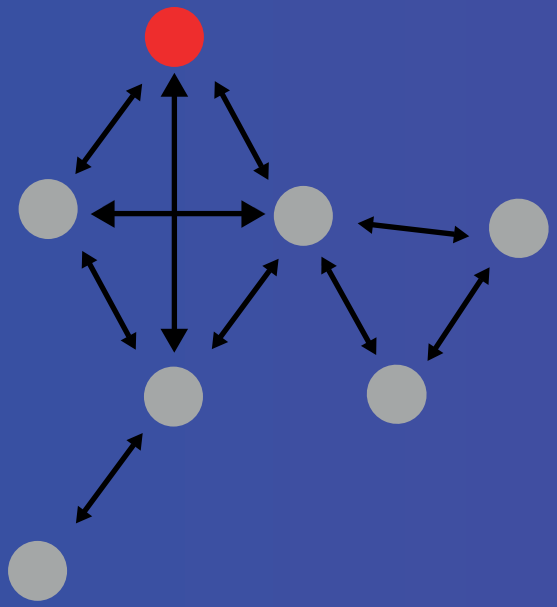
The user input and the elements have a solid property and therefore cannot break through material,
which is used in my app to create more space for content.
What is consistent, however, is the overlaying of different planes,
which do not slide through the others but always remain in the intended position.

Attributes
Behaviour

Surfaces can behave in multiple ways, in my app they follow the rigid path. Rigid surfaces stay the same size for all interactions,
they don't expand when you slide them or anything like that, except in the smartwatch prototype,
when you want a function to be magnified, it's done by tapping on it. They stretch so you can see more information about
the element, but most of the time, a new window pops up to get full information for example like the player.

Navigation (App)

Fully connected -> Multilevel



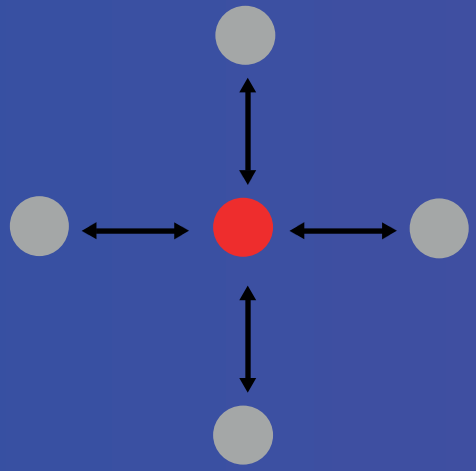
What was originally planned as fully connected, has quickly developed after further elaboration in the direction of multilevel, since it was now no longer only to switch between the main pages arbitrarily back and forth, but also under connections of the individual main pages arose. That is, if you were in this subnet of any main page, you had to return to the specific main page to get to the other main pages. This is by far not a disadvantage, because for the persona it is important to keep a certain order and therefore everything stays where it belongs. It is also not the purpose of going to a main page to use the functions of another main page. Especially since the individual sub-links of the individual main pages are often very similar or even the same. So it is also guaranteed not to overwhelm the user with too much input and functions and you can concentrate on the essentials. Some disadvantages are that you first have to go to the individual main pages to see their functions, instead of having an overview of them right at the beginning, which should not be too big a disadvantage. A small but very significant advantage is that you can always access the main function of the app from over 90 percent of the pages, which is playing music.
This navigation and the design create a very familiar environment, which is very convenient for the user to quickly find his way around.

Navigation (Smartwatch)

Sequental

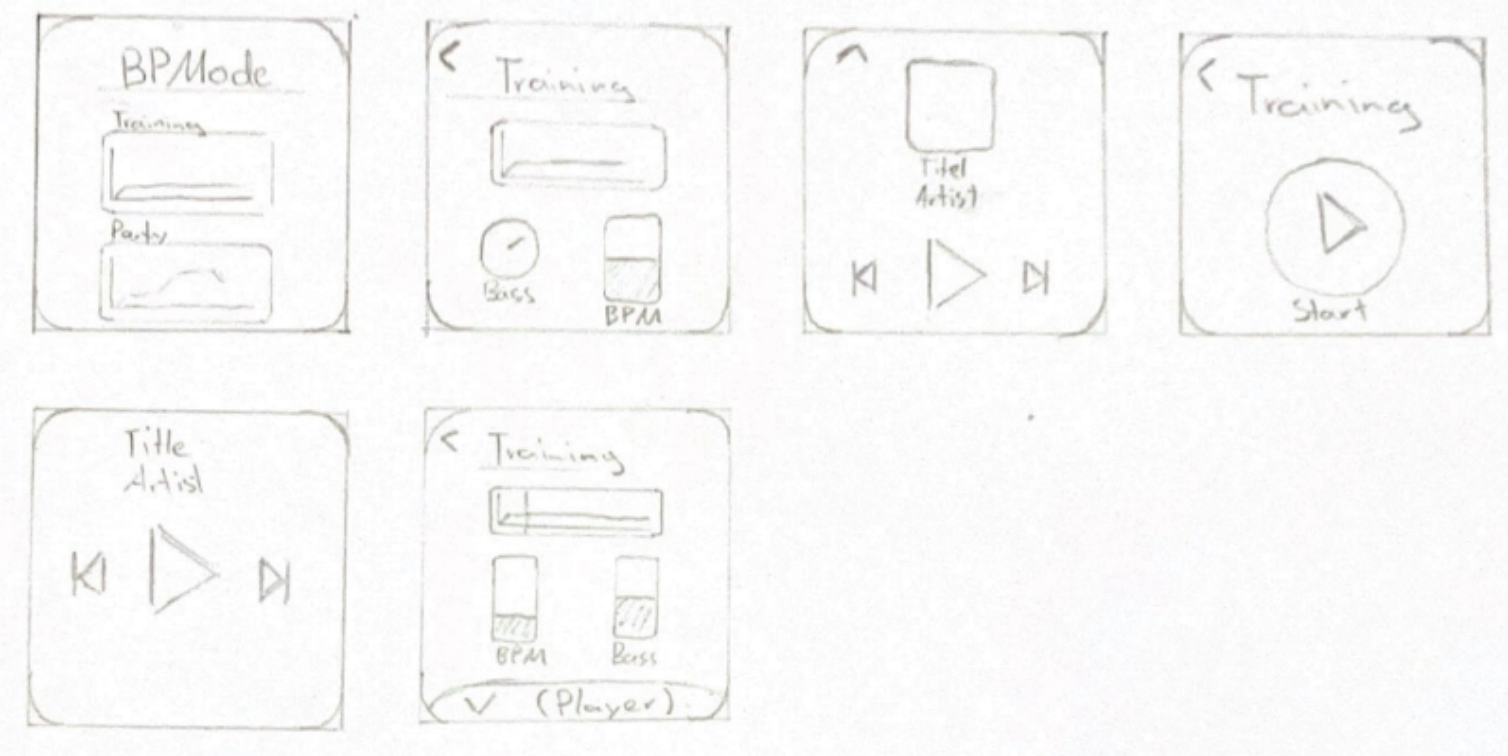
For navigation on the smartwatch, sequential would be a good decision.
It would also make sense to use the smartwatch only for the special function,
especially when jogging or exercising in general.
Thus, it is possible for the user to select different modes set on the smartphone on the smartwatch
and to play and stop music. This is because the space on the smartwatch is limited
and should not be overloaded, so it would be impractical to include all other functions.
In addition, sequental is advantageous here because you can navigate back
and forth using simple gestures without having to focus exactly on the small screen.
Of course, this also means working on functions in a minimalist way
and then adapting them to the screen accordingly in order to be able to use them accordingly.

Hub and Spoke



With hub and spoke, it would basically be like the main menu of a normal apple smartwatch,
if you don't go into detail. When you are in the main menu you see all the apps
and functions that the smartwatch has, if you now go into an app you have
to go out of it again in the main menu of the smartwatch to go into another app.
Exactly so it could also look in the app that you have all the functions
on the start page at a glance and from them you get into the spokes,
from these you have to go back to the main menu to select other functions,
so you would also have a certain order guaranteed because
you see from the beginning what you can do and from there then goes its specific way.

Scribbles



Trying methods to implement main functions

Prototype

https://www.figma.com/proto/vCT96J7no3XAmE4Mcm7Miv/Smartwatch-Prototype?page-id=0%3A1&node-id=8%3A2&viewport=524%2C280%2C0.69&scaling=scale-down&starting-point-node-id=8%3A2

Interaction choices

Of course, the main interaction is to move between the pages and use the functions
by pressing the fields provided for this purpose or by swiping across the screen.
With smartwatches, however, it is also possible to control the bpm,
bass or volume using the wheel on the side of the watch.
Hardware input and output can only be obtained through the use of headphones,
which can only be used through the use of bluetooth.
Further gestures from apple consist of putting the watch in a kind of energy-saving mode
by covering the entire display with your hand.

Design reflection

First of all, it is worth mentioning that it was a very good idea to deal with it on one's own in order to make one's own prototype of an app, which was also fun in a way. In the beginning it was hard to get into the topic, because you felt a bit restricted by the limitation of the topic to a certain kind of music app, but it turned out to be not so restricted, because you were not busy developing a whole new app, but you could get some inspiration from other apps and concentrate on the specific function. This was also a bit difficult at first, to find something that might not yet exist, but we soon came up with some cool ideas. Through the deepening and also through the previous assignments, you have dealt more with the whole design and function topic than I thought, because in the end you really had a product in front of you that really has the potential for a real app. What was interesting about the whole development journey was how detailed the whole thing actually is when it comes to constructing or deconstructing an app. A lot of things are taken for granted when it comes to using certain functions or even navigating of everyday apps. Especially when it comes to designing familiar environments for the user. We never really questioned why some apps are good or why others are immediately closed again, usually the reason was, I like it or I don't like it. But sometimes it's exactly what you've learned here, the design has to appeal to you, of course, but functions like the settings or the functions you can interact with appear in just about every app, and it's exactly when you have no idea where you're going, it is the reason for you, to close the app. So it isn't really the design in first place where I have learned a lot, it is more the idea behind the app, things I really took for granted that kind of impressed me while working on constructing or deconstructing an app.

What I'm definitely taking away is the slightly different view of the everyday apps that you use all the time, the navigation, the functions and the often recurring features that make it very easy to deal with them.