TECH BASICS II REPORT

1 I INTRODUCTION

With the goal of sharing my upcycling earrings hobby in the internet I want to set my local application from last semester on a higher level and create a working Minimum Viable Product. Thanks to Streamlit, it is now possible for everyone to work with my application on new earring creations as well as inform yourself about me and my brand handgemacht.von.svea.

2 I DEVELOPMENT PROCESS

My development process started with connecting the learned concepts to my application. This included working with a database, a KI chatbot and an API. Of course, I also thought about ways to improve my design as we are now working with Streamlit.

The database could be useful for a newsletter subscription function. In that way, I can see all the e-mail addresses for my newsletter in the database. With a chatbot and API in mind, I decided to lay this project's focus in the interactive part. In my app, this is the "craft with me" page. This page helps the user to start their own journey. The chatbot should help with quick answers to complicated questions. The API helps with images, which may inspire the crafter. I personally work with inspirational images often and want to share this approach. The crafter can type anything in the search bar and gets inspiration from, for example, beachy vibes or a chic gala.

Furthermore, I used existing Streamlit functions to my advantage. I found a navigation function as well as an expander. In that way, all explanatory text -tutorial and FAQ- could be hidden behind the expander if not needed. The navigation function helped me to achieve a cleaner look of my application and helped me organize my code in different files.

After researching the functions and connecting the learnt concepts with my application, I started drafting my application in my notes. With that base I started coding the easier pages, like the "get to know me" page or the "gallery" page. I then started with some logic and developed the randomizer page. The base was laid by Tech Basics I code, which had to be converted towards my use case. In that manner, I worked step by step towards the more complicated functionalities of my application like the decision help. I "recycled" most of the code for the chatbot, API, and database connection from class code and only changed minor things. One example: I wrote the chatbot answer in Markdown. Using the "st.write" function, Markdown formatting appeared in the answer, but when using the "st.markdown" function, The text appeared in bold, italic et cetera.

The last steps were unifying my design and improving smaller functionalities with my gained knowledge. Lastly, I added a start page with a tutorial. Then, the application was ready for testing!

3 I CHALLENGES AND LIMITATIONS

The first challenge was the design of the sidebar. I wanted to add my logo as well as a navigation. No matter what I did, the logo would appear under the navigation. After trying a lot of different things, I found the option of adding a logo (st.logo). It would place a logo at the upper left corner. I would like to display the logo bigger, but it is already presented in the biggest scale. In this case, the Streamlit function st.logo set a limitation.

The second challenge I faced was arranging a grid design with columns on the "get to know me" page. My columns were different heights. Therefore, the different vertical sections did not start at the same height, but the height the last column ended. By working with containers and columns, we could arrange a nice grid design. (Thank you, Sarah!)

The next challenge came with the arrangement of the chatbot. Hugging Chat only worked sometimes. The latest status is a working chatbot! But just in case, it might not work, I added a note, which is shown while generating the content. It says that the servers might be overrunning, and an answer can't be guaranteed. I also added frequently asked questions as well to give some help if the chatbot would not work.

The last Limitation is an error, I can't explain. I connected MongoDB to my app and already added some e-mail-addresses to the database via the sign-up page. It all works in the rooms of the HMS, but if I work with the app in any other place, it is not possible to write data into the database. I did not change anything in the code. In case of a not working newsletter sign up, I added user feedback which links my Instagram page and a note to get in touch with me there.

4 I FEEDBACK FROM TESTERS

I asked three people to test my app on usability, design, and accessibility without any further information given by me. In the following, I will present the feedback as well as my solutions to it.

4.1 I Usability

Although my first tester does not speak English very well, she perceived the usability as intuitive and easy. She enjoyed the interactivity of the application and especially liked the "decision help" as well as the "craft with me page" due to its usefulness. My second testers feedbacked an easy handling and navigation of the app. The purpose of the pages was easy to understand and the structure user-friendly. The third tester liked the clear arrangement of the sections as well as the use of white space for structure. One criticism is that the logo does not lead to the start page. Although the start page is linked to the navigation, it caused confusion. This feedback was underlined by my first tester, which also tried to navigate through the logo.

As a solution, I found out about the option of linking a URL to the logo. By linking to my deployed page, I can achieve the functionality of a logo leading to the start page. Sadly, this function opens a new tab and does not navigate within the same tab. Therefore, I can't improve the usability of the navigation due to Streamlits restricted functions.

4.2 I Design

For the design, I received solely positive feedback. The colors underline my brand, the simple design leads to an easy orientation, and the pages seem pleasant to the eye by not being overloaded. The Design is understandable and consistent through the whole app.

4.3 I Accessibility

Regarding accessibility, the easy navigation though the pages underlines an accessible app. The only criticism is that the font size might be too small for some users. Especially the tabs on the "craft with me page" can be overlooked easily.

Towards this problem, I sadly could not find a way to change the size of the font. But I do not see that much of a problem in that. As a last resolution, the user could adjust the screen size with "strg"&"+" to the size they like it.

5 I CONCLUSION

I enjoyed the process of developing the application step by step and improving it with each new functionality I learned about Streamlit. I was able to learn new ways of working with python as well. The user testing underlined the importance of showing your work to people who do not know your intention. I lead me to improving my application even more. I now have my very own Streamlit project, which is built up on the learnings of the Tech Basics II class and contributes to sharing my favorite hobby of upcycling jewelry.

6 I SOURCES

6.1 I General sources (often used)

Markdown Text: https://docs.streamlit.io/develop/api-reference/text/st.markdown
White space: https://discuss.streamlit.io/t/create-empty-space-to-separate-portions-of-the-app/8689
Container width button (used on all buttons): https://docs.streamlit.io/develop/api-reference/widgets/st.button

6.2 I Specific sources with code line numbers

Navigation: https://docs.streamlit.io/develop/api-reference/navigation/st.navigation
→ entrypoint.py, ll. 13-25
Sidebar: https://blog.streamlit.io/designing-streamlit-apps-for-the-user-part-ii/
→ entrypoint.py, l. 7
St.logo: https://docs.streamlit.io/develop/api-reference/media/st.logo
→ entrypoint.py, l. 10
Embed calendar: https://discuss.streamlit.io/t/is-it-possible-to-embed-an-external-website-in-the-streamlit-app/54197
→ events.py, l. 30
Expander: https://docs.streamlit.io/develop/api-reference/layout/st.expander
→ start_page.py, ll. 29-79 / craft_with_me.py, ll. 91-110
Radio buttons: https://docs.streamlit.io/develop/api-reference/widgets/st.radio
→ decision_help.py, ll. 14-39
St.link_button: https://docs.streamlit.io/develop/api-reference/widgets/st.link_button
→ sign_up.py, l. 38 / craft_with_me.py, ll. 28-33
Play music: https://docs.streamlit.io/develop/api-reference/media/st.audio and craft with me music: https://pixabay.com/de/music/schlagt-dreaming-through-dusk-lofi-beats-281206/
→ craft_with_me.py, l. 129
Change colors: https://docs.streamlit.io/develop/concepts/configuration/theming
→ .streamlit/config.toml

7 I STREAMLIT APPLICATION LINK

https://handgemacht-von-svea.streamlit.app/