**Report Tech Basics II - Cross country skiing application**

Improving my Tech Basics 1 application this semester was a way for me to see what I learned in the last two semesters. For some of the pages I used my code from the first version but had to adjust it heavily. My focus for the second version of my application was to see what I could code on my own. Therefore, I decided to implement different pages that were more complex and challenge myself.

The process was the following: Develop the structure of the app, meaning everything should work. At this point I didn't change the colors or focus on the placements on each page. I decided to build each page separately and bring them together one by one. I divided the pages into different levels (easy, medium, hard) and depending on the level changed my coding approach. Easy pages such as the *Information* page, the *About me* page or the homepage were built in one swoop and used to have a success if I was stuck at the more complicated pages. Hard pages such as the *Weather and Waxing* page and the *Training sessions* page (including the login and registration) weren't built in one session and I split them up into smaller blocks. Whenever I felt like I wouldn't make any progress and was stuck in my thought approach I decided to work on another page and save this for later. Additionally, I reused seminar code for the *Training sessions* page and worked with ChatGPT for the *Weather and Waxing* page to make it work. For the level medium pages, I was confident that I could realize them without too much effort compared to the hard pages but wasn't sure if I would still need help if I'm stuck. I always started off with trying to implement the medium level pages in one go but if there wasn't any progress I decided to take a break and work on another page.

Only after the structure was standing, did I revise the design of the different pages with the time I had left. In the end there was enough time to change the design according to my ideas. This included changing sizes of text, thinking about useful colors for headers or text, using a config.txt file to change the color scheme of the app, including more pictures and changing the placement of different elements.

While it is much easier to make the design look good with Streamlit there are some limitations to it as well. While tkinter.place() enabled precise placement this was harder with Streamlit. It was possible to adjust the placement with containers or columns but only up to a certain point.

Another challenge was the login and registration. The initial plan was to do it for the whole application. However, this wasn't working due to a circular import. In this case I could find a suitable solution and it wasn't too bad that I encountered this error since this made me rethink the usefulness of a login for the whole application and I decided that it is only useful for the

*Training sessions* page and could be a hindrance for people to use my app if it was necessary to register to use the application at all.

The biggest challenge however was probably the *Weather and Waxing* page with two different API keys. Since the page didn't progress at all and there were error messages each time that I tried it out, I decided to work with ChatGPT to make this page work. The result is the page that I imagined but, on the downside, I struggle to understand some lines of the code and I couldn't find out what my own mistakes were for the previous versions. The working page was my main goal and the result contains everything that was supposed to be incorporated but it still isn't good that I couldn't understand my own mistakes since it prevents me from becoming a better coder.

After the app was deployed, I collected user feedback for the report so I would have the opportunity to improve my app further in the future. Test users were asked to give feedback regarding the three aspects design, usability and accessibility. Additionally, they were asked to inform me about any errors while using the app and if they had additional comments that wouldn't fit in any of these boxes.

Coming to the aspect of accessibility, the biggest reported problem is the chosen language. While English was understandable for the younger test users, it affected older user who can't understand much English. A translator function that provides different languages would be a solution to that problem. Other than that, my users weren't able to give much more feedback from their own point of view regarding accessibility. That is why I used a color contrast checker to see if there is enough contrast in the text and background colors and see if the app is accessible for visually differently abled people. The result is that the contrast is very good.

For the design there were some main points referred to by the users. Firstly, the picked color scheme resonated well with them. A user mentioned that they think of the colors white and blue first when it comes to skiing. Another point that was referred to more than once is that the application seems overloaded with text. The users mentioned that they would prefer an app with less text and more interaction possibilities since there is a big resemblance to a website for the current app. The feedback for the *Motivational quotes* page was very positive. Users liked the simplicity and used design elements such as the rainbow header and the picture. However, contrary to that the *Weather and Waxing* page was criticized since there are too many elements on this page and this led to having to search for important information. All in all, it can be said that there are some positive as well as negative aspects of the implemented design.

Lastly, coming to the aspect of usability the feedback that the *Weather and Waxing* page is too stuffed is relevant for this as well since it is a problem when users can't figure out a page easily. One user mentioned that you could trick the quiz and rule out answers if you would skip the questions and press "Show result". Since a) is the auto-pick and is never the right answer this makes it possible for the user to only choose between b) and c) since a) is never correct. Other than that, the feedback regarding the usability was really good. Even the non-English speaking people could use the app well without help from others. Their feedback was mostly similar: most of the pages where they could do something they figured out in an instant, only for some varying individual elements they had to try how to use it.

One thing that doesn't work out as planned is the city input for the weather. While testing, users found out that there would appear an error message if they entered a fictional, non-existing city or if the input field was empty. Fortunately, the error disappears once an existing city is entered but it still is a malfunction of my application. Additionally, in other cases, nonsense inputs like "Hi" or "123" would display the weather for this entry. Lastly, it is not clear if the city that was entered is the city that you meant. For example, Berlin is the capitol of Germany but also a city in New Hampshire, US. All these errors for this input field were found out during testing phase but other than that no errors occurred during testing.

To sum up, my MVP did quite well for most parts, based on the user feedback as well as my own tests after deploying it. However, there are some things that I want to change in the future and I would like to elaborate on two ideas for further improvement. Firstly, I want to remove the weather forecast including location settings, current weather and daily forecast. There are currently too many mistakes happening with the city input field which could make the application seem faulty even though it is only one field that isn't working correctly. However, once this is fixed, I would like to include it again but change the design and use different tabs for the page to be less overloaded. These changes are needed for this page to be on the same level as the other pages. Secondly, I want to tackle the "too much text" problem. The different pages have too much text as users also criticized and I want to include information-buttons. This would mean that only people that decide to click on the button see more text, making the pages less text heavy.

Overall, developing the second version of my cross-country skiing application was a valuable learning experience that highlighted my progress and as well as areas for improvement. The feedback enabled me to see the application from another perspective and helped me to refine my app, ensuring that it becomes even more user-friendly, accessible and engaging.