Report Tech Basics II Exam

Name: Anissa Blonn

Student-ID: 3048204

Module: Technological Basics II

Submission date: 15.03.2024

Introduction:

For this exam project, I further developed my app “ConcertBuddy” from the Tech Basics I exam project and built the MVP for it.

To explain it shortly the app “ConcertBuddy” can help users find others who will go to the same concert as them, so if you do not want to go alone to a concert and would like to meet new people with a similar music taste this is the app for you. The whole concept of the app is inspired by dating apps like Tinder or Bumble, especially the way you find new people by swiping left or right and having matches with people who have mutual interests. There are also other concert-related features that users can benefit from. Having more experience with Python now, I was able to implement more features that I could only theoretically explain in Tech Basics I.

Methodology:

To determine what I wanted to do with my app and what further features I could implement, I first looked at my first version of the app and what ideas I had in the corresponding report, furthermore, I looked at the dating apps where I got the “matching users” idea from and drew some more inspiration from them regarding what people should answer in their profiles. I also asked myself and some of my friends what kind of features would make sense in a concert-related app. That is how I got the ideas for the main features. Some ideas and ways of doing things I got while programming the app, and when I thought about the things we did during the Tech Basics classes and how I could use some of those things in the app, and what parts could be necessary to build my app. This way I got the idea to implement the calendar to determine the user's age, as it is something we did in class and something I have already seen other applications in real-life use. I also thought the distinction between a new user and a returning user would make sense and implemented this in my code as well as the chatbot feature as a visualization that users can interact with one another through the app. For other parts, I mostly

used more of the experience of building GUIs I gathered in class, sometimes I needed to do some research for my ideas and additionally go through a lot of trial and error.

Design & Limitations:

In the following, I will explain the technical and design aspects and the features implemented in the app (and MVP) as well as which struggles and limitations there are.

The app starts with the homepage where the user can decide if they want to sign up (if they haven't used the app before) or log in (when they are a returning user). The information the user entered by signing in or logging in will be saved as data and therefore it can be checked if a user already exists or if a username is already taken. There are if the user has signed up or logged in, six main pages easily switched in between with the icons on the bottom for a better user experience.

Starting at the profile page, here the user can enter personal information about themselves, like their age, gender, education, languages they speak, and their location. Furthermore, they are able to select prompts about their music taste and enter an answer, to show others their taste in music. This concept was also inspired by Bumble. To select the age, the user has to select their date of birth in a new window that pops up when clicking the button and closes when the user has selected their date of birth and confirmed it. The calculated age will then be displayed in their profile (this only works correctly for birthdates after 1999 and I was not able to fix it) and the window will close. Getting this working took some time and research but in the end, I was able to find a working way as I really liked the idea of selecting the date with the calendar. Some of the buttons on this page have no command because I am not able to take data from external websites. Additionally, the entered user information does not get saved in their profile as this would probably require the use of a bigger and more complex database system.

On the next page, the user is able to keep track of their concerts and can add new ones that the hypothetical algorithm of the app will then use to find other people going to that same concert. Here I was struggling with saving and working with the data as I would say it is not my strong suit but I think it is important to show the workings of the app. After doing some research and trying out a lot of things I overcame the returning error messages and found a way. However, I was not able to find a way to only display the table with the concerts if there is concert data available. The page for adding the concerts is nearly the same as in my Tech Basics I project, with the addition of the dropdown menu for the seating or standing category and a button that saves the entered information in a .csv file for further usage like in the table.

The next two pages only show images of how the page should look, because I was not able to implement a working map and I do not have the concert data for each concert happening somewhere. To get this site working I would need this kind of concert data from other sites, for example from ticket-selling sites like Ticketmaster or Eventim. This page in the app could then provide a direct link to these sites where you can purchase a ticket and it could also show concerts where other users are selling tickets for. I imagine this to be something the user can filter as well as the genre of music, the timeframe, and the location radius.

The actual page where the user gets shown other profiles is again just an image of how it would look like because again I do not have other users' data I can work with and am not able to combine it into a profile page of the sort I imagine the app to have. But to make clear how it should work I designed the mock-up in Canva and placed it as an image on the page.

For the chat page, I used the idea of a chatbot we did during one of the Tech Basic II classes and added the user Mark as an example of how users can communicate with each other and to give a visual idea of the purpose of the page. Of course, it is only a simplification of how I imagine the chat page to look as there would be more users and different chats for different users and obviously real people replying to the user.

Lastly, there is a page for a ticket sale, here users can choose if they want to sell a ticket themselves or if they want to buy a ticket from another user. For the selling a ticket page the user can choose for which one of their concerts they want to sell their ticket and some more additional information like the selling price of the ticket. Further functions are not yet available because I do not have the data for this.

Creating all the missing data to then have data to implement these missing sites I thought would be too complicated for this project. However, with the parts I could implement with the provided limitations and my research the MVP for my app shows the potential users what the app will look like and what features users will have at their disposal to make their concert experience better.

Video to show the design:

https://youtu.be/Uuve94V4DSo

Sources

Line 972, 973 (Frame)

Kb. n.d. “Inf-schule | Gestaltung Des Layouts » der pack-Manager.” https://www.inf-schule.de/software/gui/entwicklung_tkinter/layout/pack.

“Align A Button To The Bottom, Using Tkinter.” n.d. Stack Overflow. https://stackoverflow.com/questions/48723923/align-a-button-to-the-bottom-using-tkinter.

Line 138 – 160, 232 – 257, 392 – 413 (Dropdown Menus)

GfG. 2020. “Dropdown Menus Tkinter.” GeeksforGeeks. November 26, 2020. https://www.geeksforgeeks.org/dropdown-menus-tkinter/.

Fabio Musanni - Programming Channel. 2023. “Style Tkinter Dropdown Menus in Python | OptionMenu Change Default Icon in Tkinter GUI.” https://www.youtube.com/watch?v=TmVTpHJ6lnc.

Line 86, 818 (Lambda Command)

GfG. 2020. “How To Bind Multiple Commands To Tkinter Button?” GeeksforGeeks. December 26, 2020. https://www.geeksforgeeks.org/how-to-bind-multiple-commands-to-tkinter-button/.

Line 86 (Close the Calendar Window)

Kb. n.d. “Inf-schule | Fenster und Dialoge » Fenster Öffnen und Schließen.” https://www.inf-schule.de/software/gui/entwicklung_tkinter/fensterdialoge/oeffnenschliessen.

Line 12 (helpers.py) (The add_image problem as explained via E-mail)

chee wan go. 2020. “Solved: AttributeError_ Type Object ‘Image’ Has No Attribute ‘open.’” https://www.youtube.com/watch?v=f6gbKhvyPi8.

Line 289, 294, 527 (Data Specification)

Grupman, Celeste. 2023. “Python If Else Tutorial: Control The Flow Of Your Code.” Dataquest. March 6, 2023. https://www.dataquest.io/blog/python-if-else/.

pack function Features

Kb. n.d. “Inf-schule | Gestaltung Des Layouts » der pack-Manager.” https://www.inf-schule.de/software/gui/entwicklung_tkinter/layout/pack.

“Python-Kurs: Layout-Manager und Layout-Management in Tkinter.” n.d. https://www.python-kurs.eu/tkinter_layout_management.php.

Line 270 – 296 (Display the Concerts)

plus2net. 2022. “Tkinter Treeview Displaying Data Rows With Headers From Different Source Like CSV File & Excel.” https://www.youtube.com/watch?v=vpmMNh4dJZ0.

Line 524 – 527 (Add the Concert Data to the Dropdown Menu)

“Python Tkinter Import Csv Data To OptionMenu.” n.d. Stack Overflow. https://stackoverflow.com/questions/63813970/python-tkinter-import-csv-data-to-optionmenu.

Line 698 – 701 (Configure the Textbox)

GfG. 2022. “Python Tkinter Text Widget.” GeeksforGeeks. January 24, 2022. https://www.geeksforgeeks.org/python-tkinter-text-widget/.