2025

FITLISTIC

HOLISTIC FITNESS

Fitlistic V2 - Project Report

TECH BASICS II (STREAM B), SARAH HAQ
FINN KÜNZEL | 4000323

LEUPHANA UNIVERSITY | February 28, 2025

Introduction

With the idea in mind to create an app that creates a bridge between physical fitness and mental wellbeing, the prototype of the Fitlistic application was created. While this prototype offered a clear view of the possibilities and the structure of an app of this kind, it was still missing the function to make it a fully working product.

This report shows how the prototype developed with the Python library Tkinter was transformed into a fully working Streamlit application. It will shed light on the design process, the limitations encountered, and how most of them were overcome in developing this fully functioning product.

Development Process

The development started with a close look at the prototype and a deeper dive into what can be reused from the prototype and what things need to be redesigned or added to help transform it into a fully functioning application. Design-wise, the prototype offered a well-thought-out color palette and structure that could be reused for the development of the new MVP version of the app. The green and blue accents from the logo combined with a clean white look created a pleasant atmosphere around the topic of fitness and wellbeing. Feature-wise, the prototype offered a view into what functionality the app should offer. It should have the possibility of tracking and monitoring a wellbeing score. The chance to complete detailed holistic workouts and a reminder function to create reminders for workouts.

While these things demonstrate the potential of the Fitlistic app, the actual application needed more functions to achieve the state of a usable app. The following things needed to be added. First the ability to register and select goals that affect the app experience. Then the reminders need to be saved in the app. The possibility of chatting with an AI about holistic fitness is also a neat feature that should be added. A workout creator that creates a tailored holistic fitness plan makes the app a real fitness app and An actual wellbeing score and chart to display it that is generated from the user logging his wellbeing is making it possible for the user to track is wellbeing.

After the to-dos were clear, the development began by doing the basics. In this case, a MongoDB database was set up and connected to the Streamlit Code. With the database set up, the next focus was migrating the existing application into Streamlit. This part went smoothly, and Streamlits built-in theme configuration options made creating the prototype's light blue and light green style easy. The next step was the register and login process. For the registration process, the option to enter the height and weight and choose from holistic fitness goals was implemented. Afterward, the function to check if a user is logged in before entering a page was implemented. For this purpose, an annotation pattern was used. From here on it was time to fill the existing pages with real function.

At first the Homepage was added with the possibility to log the wellbeing and displaying it in a chart after five entries were made. From here on the next part was to add a new page for the AI-Coach. With the knowledge from the sessions and a look at the OpenAI documentation the development of the AI-Coach went smoothly. While the AI-Coach should offer general help there also should be a non-AI workout creator that creates a useful holistic fitness plan. The next part was to add different exercises to the MongoDB so a workout creator can use these added exercises to create a user specific holistic fitness plan. For the generation ChatGPT was used to generate exercise Json documents that could be imported into the database. Then the workout creator could be implemented. Here the exercises from the database collection are fetched and are combined with a workout that is part of a seven-day plan that gets generated. The generated workout plan gets also saved as a collection per user and can then be shown on the Exercise page. Here the user can complete the days and see the instructions to the selected exercises from each day again. After this was done the reminder function was programmed to create reminders and save them in the database. The last thing that was added was the option to change details about the profile on a specific profile page.

Limitations and Solutions

During the development process I faced different challenges that required different solutions. The first questions came up while working on the registration process and thinking about a good validation for the forms that hinders the submit and also gives useful feedback at the fields where the errors are. This topic could be cleared while asking for support in class and getting the hint to look again at the GitHub code from the lesson where we created a registration form together [1].

While looking for a solution to hide the option to make an image fullscreen I stumbled across the Streamlit discussion platform where a user named AvratanuBiswas explained how CSS styles can manipulate the Streamlit data [2]. This is something I used to hide the fullscreen option but also later for example to style the sidebar and hide the login and register pages from it after a user logged in.

In general, most of the problems that I encountered while building my prototype could be resolved by asking in class, researching various sources, and trying things out.

User Feedback (Usability, Design and Accessibility)

Internal and External References

Looking back into things we covered in class

[1] S. Haq, "Tech Basics Two," GitHub. Internet: https://github.com/shaq31415926/tech-basics/tree/main/tech_basics_two

Accessed: Feb. 27, 2025.

Hiding things in Streamlit through CSS – Answer by AvratanuBiswas

[2] AvratanuBiswas, "Hide Fullscreen Option when displaying images using st.image," Streamlit Discuss. Internet: https://discuss.streamlit.io/t/hide-fullscreen-option-when-displaying-images-using-st-image/19792

Accessed: Feb. 27, 2025.

Looking into how to use OpenAI API

[3] OpenAI, " Developer quickstart" OpenAI Platform. Internet: https://platform.openai.com/docs/quickstart

Accessed: Feb. 27, 2025.

Rating Component in Streamlit – Component by Tian

[3] Tian, " New Component: Star Ratings!" Streamlit Discuss. Internet https://discuss.streamlit.io/t/new-component-star-ratings/36829

Accessed: Feb. 27, 2025.

Listing different exercises with image and text (Lines 41-54 & 294-310)

[4] OpenAI, "Response generated by ChatGPT (GPT-4o) on how to list a combination of images and text in an efficient way in Tkinter," OpenAI. Internet: https://www.openai.com/chatgpt

Accessed: Aug. 8, 2024.

[Prompt: "Create a simple and clean code coherent Python Tkinter script that displays a list of exercises. Each exercise should be shown with a small 50x50 image on the left and a description on the right. The images should be resized using LANCZOS resampling to fit the specified dimensions and should be displayed alongside their corresponding exercise descriptions."]