

Drawing idea generator documentation

For my project, I decided to build a drawing idea generator because I have always loved drawing and thought this would be a fun application that I personally would use. There are apps and websites like that out there, but building it myself comes with the benefit that I can customize it to my own liking, especially use my own lists of words for generation to match my personal interests.

I started by creating the very basic functionality of generating a prompt by randomly pulling words from lists. Then I built a simple GUI around that. I continuously added new features and updated and improved the GUI and its layout in between. First, I added the options of generating a random material and style in addition to the drawing prompt. This was quite simple, because these additional options are also pulled from lists of words randomly. A bit more complicated was implementing them into a GUI in a way that allowed for selecting and unselecting options separately. I also really wanted to include randomly generated colour palettes, because this is something that I generally enjoy a lot. This turned out to be more challenging, because I wanted to generate random RGB values and not just use the colour names that are integrated into python. For this purpose, I wrote a function that generates random RGB colours. Then I learned that tkinter doesn't support RGB, but luckily I found a function that translates RGB into Hex values. Still, this was more complicated than the other generated options, because the colours needed to be displayed as colours and not as text and I also wanted to be able to choose between different numbers of colours that are generated. However, after a little bit of back and forth, this worked quite nicely.

By far the most challenging part for me was integrating data that is served in a csv file, because I have never been very fond of working with data frames. Still I thought it would be nice to be able to save certain generated ideas and view them later. Originally, I also wanted to tie this to different user accounts. This was quite hard for me to integrate though, so I decided that the ability to create and log into accounts is not that useful for my app compared to the effort that I needed to put in in order to make that work. I still wanted the option to save ideas though. This definitely took me the longest to get it to at least somewhat work.

I then decided to also add the option to generate landscapes instead characters. Personally, I mostly draw characters, so landscapes were not a top priority for me from the start. Still I figured that my app would benefit from offering a bit more variety. Therefore I added landscape counterparts to most of the existing widgets and functions. This caused the code to be very long and a bit confusing and there would probably be a better way to do this, but I was hesitant to change my existing code too much in fear of breaking it.

For the design, I decided to use fun pastel colours, because I like the playfulness that they convey. The app is meant to be fun, so I wanted it to look fun as well. For some of the text, I used Comic Sans, because I feel like it fits the playful vibe as well. I also used images with hand drawn text or symbols for some of the text as well as the background, to give the design a more unique and personal touch.

Displaying the saved ideas definitely took the longest to implement and while it works now, the feature is not perfect. You can open the favourites page when the saved_ideas file is empty, which will show an empty page basically. Also there is no visible indication for how many ideas are saved or when the last idea is reached, other than the next idea button won't have an effect anymore. Additionally, I meant for the favourite button to be replaced with a non-functional variant when an idea is saved to stop the user from saving the same idea twice. However, I could not think of an easy way to do this without room for error, so I skipped this option. And lastly, when the colour palette for an idea contains less than five colours, I had to replace missing colours with 'white' in order for the code to work. Therefore, these colours still show up as white squares. None of these circumstances are ideal, but I think the app is usable enough regardless.

There are not many possible features that I could think of that are not there yet and that I would like to have. Maybe I would enjoy the option to create a prompt with more than one character from the list or the ability to upload my own drawings that I have made using the generated ideas.

I haven't had the chance to properly test the generator and draw something based on its suggestions yet, but I'm looking forward to hopefully being inspired by it and also expanding the word lists.

References

Lines 40 – 53, `page_frames = ... i.grid_rowconfigure(3, weight=1)`, source: ChatGPT

Line 194, `return f'#{r:02x}{g:02x}{b:02x}'`, source:

<https://stackoverflow.com/questions/51591456/can-i-use-rgb-in-tkinter>

Lines 398 – 401, `if idea_number < 0:`

`idea_number = 0`

`elif idea_number >= len(saved_ideas):`

`idea_number = len(saved_ideas) – 1`, source: ChatGPT