

German Vocab Randomiser App

Links

Code Style Guide

GitGub

Please find the source code here.

https://github.com/anna-cee/German_Vocab_App

Trello Board

Purpose & Features

The German Vocab Randomiser App is a simple CLI app tool designed for the user who is a language learner to learn, recall and recycle vocabulary from a pool of lexical items targeted at their level. The lexical items are organised by topic, with a translation in to English and an example sentence in topic. The app features are structures sequentially in order of complexity for the learner. In German_Vocab_Randomiser App you can:

- Choose a topic from a menu.
- Get a randomised list of 10 items with a translation and example.
 - the following steps all practice this list.
- Generate on screen flashcards for each list
 - Learn the vocab items in either language direction
 - Unlimited cycles available, move on at end of any cycle.
- Generate a matching quiz to recall the items
 - Spelling also practiced here
 - No penalty for incorrect answers or spelling, unlimited guesses
 - Option to request answer
 - First-time guess score given!
 - Unlimited cycles available.
- Generate a gap-fill quiz to recall words and identify meaning from context.
 - Spelling practiced.
 - One guess only, then answer given
 - Unlimited cycles through quiz available.
 - Exit or move on at end of a cycle.

For a detailed rundown of the concept, features and code, please see my slide documentation in /ppt.

Requirements

Python3

Python3 is required to run the app. Check if you have python3 installed run:

```
python3 --version
```

You may have previous versions of Python. Upgrade to Python3 for this app.

To install python3 according to your OS, go to <https://www.python.org/downloads/> and use the official installer.

PiP Install

Pip is a widely-used package installer for Python. A prompt in the bash script will get Pip to install all the app's dependencies automatically when it opens using the requirements.txt. Pip is included with Python versions 3.4 and higher. So after installing Python3, check if you have Pip on your system.

```
or pip3 --version
```

If it's not showing or you're not sure, you can install pip.

```
pip install pip==1.0.2
```

Or to check out Pip before downloading it, go [here](#) for more info

German Language Keyboard

You'll be able to use all the features with a standard keyboard, however some user input requires unique German alphabet characters to return a correct response. So, any words with ö, ä, ü, ß

For Windows

- Set system language to German: click on language bar and select German, or for more info go [here](#)
- Open Onscreen Keyboard, for more info go [here](#)

For Mac OS

- Set system language to German. Go to System Preferences > Keyboard > Input Sources and add German. Ensure box "Show input in menu bar" is ticked. Or for more info [here](#)
- Then in menu bar ensure language is set to "DE", then click on icon and select "Show Keyboard Viewer". Or for more info [here](#)

This is the most straightforward and error-free way to enter the characters. It also allows you to learn your way around the German Keyboard layout.

Dependencies

All package dependencies beyond the standard library in Python 3 will be installed by PiP when the program is run. They are listed in requirements.txt. Instructions for installing PiP is in the *Requirements* under *PIP*.

```
colorama==0.4.6
iniconfig==2.0.0
packaging==23.2
pluggy==1.3.0
prettytable==3.9.0
pytest==7.4.3
wcwidth==0.2.8
```

Installation Guide

- Before installing, ensure you've checked your Python version as per *Requirements*.
- Get a local copy of the repo:
 - Go to the repo [homepage](#) on GitHub
 - Click the green Code button to open a drop-down menu and choose "Download Zip" to copy the app files to your local drive.
- Unzip and go to code on your local drive
 - Open your terminal and navigate to your Downloads directory (or folder where your web downloads are located).
 - Locate the zip file there and unzip. The name could be German_Vocab_App.zip or German_Vocab_App-main.zip To unzip:
`unzip German_Vocab_App-main.zip`

- Once open navigate into the main App file.
cd German_Vocab_App-main
- Then into the source code file
cd src
- Run the code..
 - From here you can run the following command to get the bash script running, which will load the python program.
bash ./run_vocabapp.sh or ./run_vocabapp.sh
 - If permission problems are present, run the following command:
chmod +x run_vocabapp.sh
then run the file again... bash ./run_vocabapp.sh
 - The program will load and go from there!

Style

Code Style Guide

Sources

Garrels, M (2008) *Bash Guide for Beginners*, viewed 26 October 2023, <https://tldp.org/LDP/Bash-Beginners-Guide/html/index.html>

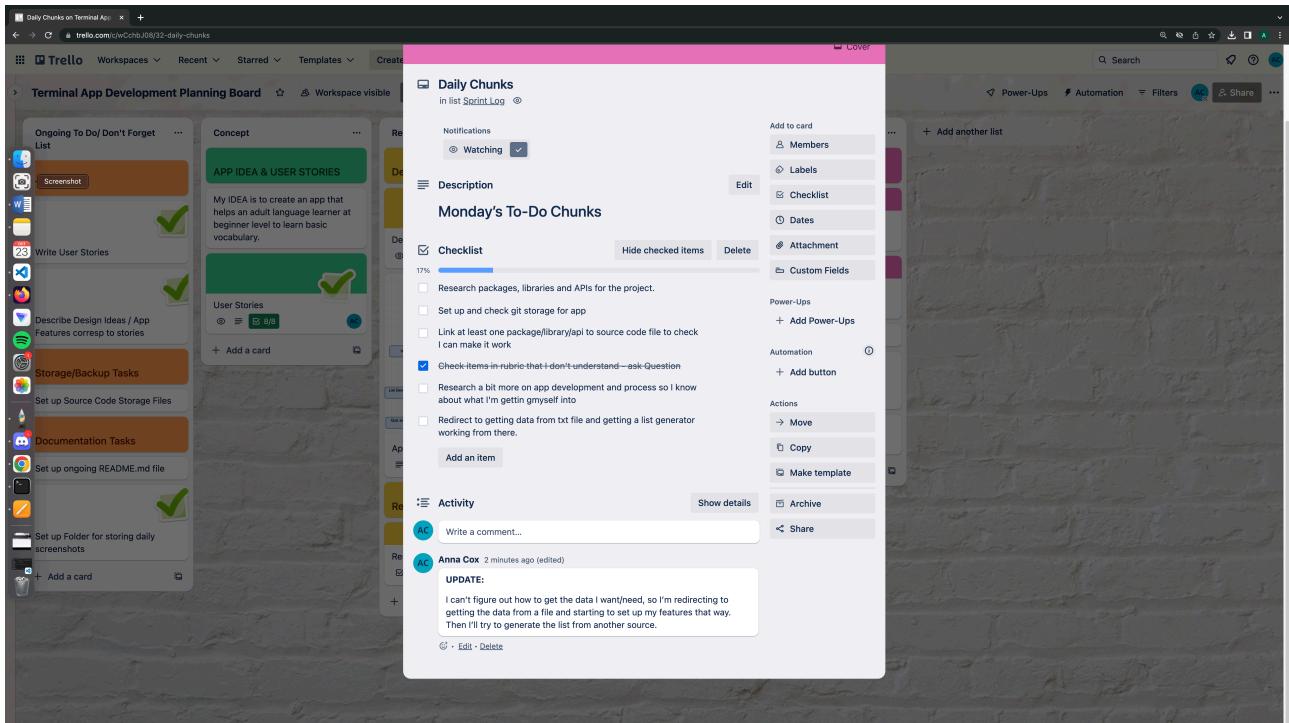
Goethe Institute (2012) *Goethe-Zertifikat A1 Start Deutsch 1: Wortliste*"Goethe Institiue, viewed 25 October 2023 https://www.goethe.de/pro/relaunch/prf/de/A1_SD1_Wortliste_02.pdf

Software Development Plan

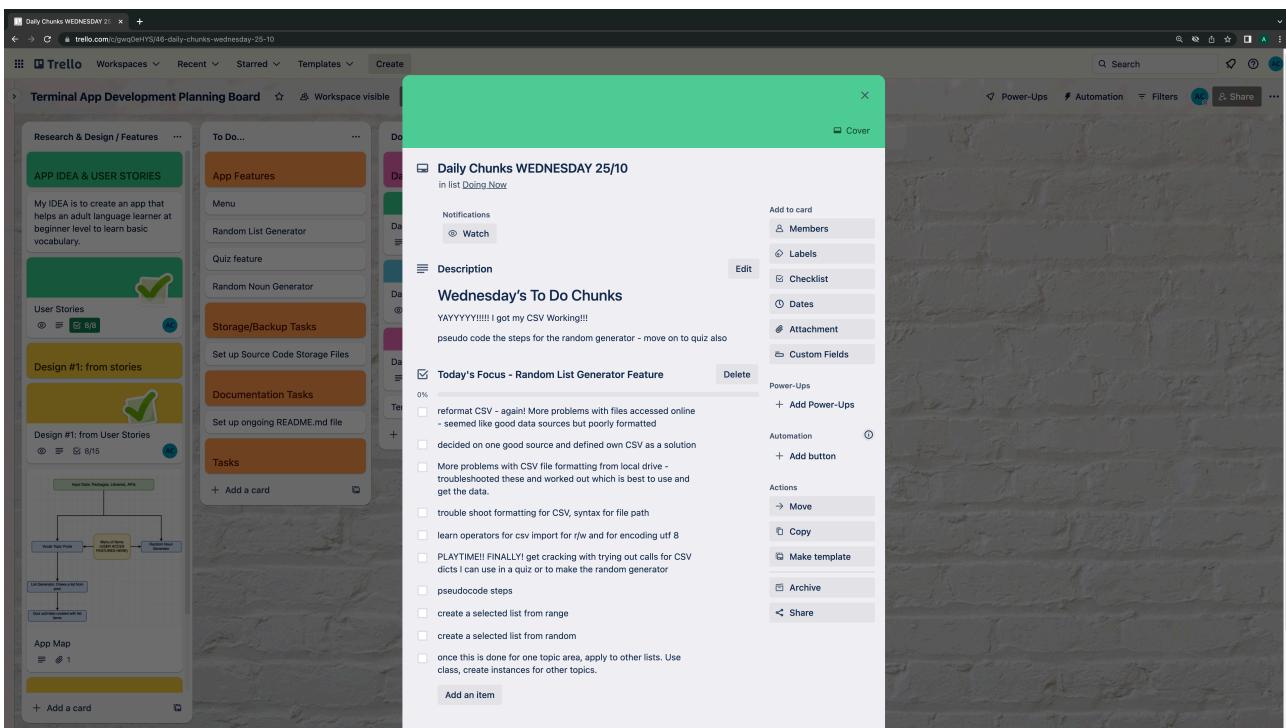
Please see Trello board for more.

Please see slides for more.

Monday 23.10



Tuesday 24.10



Wednesday 25.10

Wednesday's To Do Chunks

- YAYYYYY!!!! I got my CSV Working!!!
- pseudo code the steps for the random generator - move on to quiz also

Today's Focus - Random List Generator Feature

- reformat CSV - again! More problems with files accessed online - seemed like good data sources but poorly-formatted
- decided on one good source and defined own CSV as e-solution
- More problems with CSV file formatting from local drive - troubleshooted these and worked out which is best to use and get the data.
- trouble-shoot-formatting for CSV; syntax for file path
- learn operators-for-csv-import for r/w and for encoding utf-8
- PLAYTIME!! FINALLY cracking with trying out calls for CSV dict I can use in a quiz or to make the random generator
- now I can call items well and add text - try calling 10 items, then call 10 random items
- pseudocode steps
- create a selected list from range
- create a selected list from random
- once this is done for one topic area, apply to other lists. Use class, create instances for other topics.

Activity

Write a comment...

Next Feature - Quiz Generator Feature

- I have example sentences; can I use them to make a quiz feature? Let's pseudo code and start problem solving how to make that!
- #quiz generator
- #get-the-context-sentences-from-csv
- Got-It!
- #match keyword in the context sentence with the target vocab
- #can I create a checker activity for it?
- #can I block out the key word?
- #how could I do that? replace it with a _____?
- #item is returning a string or letters; can I convert that to a list of words, then match the word?
- Yes I found a split() method for converting to list of words - now can I match one word in the list with the target vocab?
- Yes I can, but the articles in my target vocab are causing a problem; I might have to use the split string on both items!
- This was possible! I could make both strings into lists of strings (words) that I could iterate over and find a match!
- Now I have the match; I can get the index #, can I use that to print the sentence with the target vocab replaced with _____?
- Yes I can print it, but it's still a list, find a way to return it to a long string; concatenate like this: target vocab, still sitting in variable storage, should be matchable with user input later even if replaced in list with _____?
- I can do this with join() method on the list again! There might be a easier way of doing this, but for now this is working well!
- Nope the variable is not stored and target vocab gets replaces, there could be a workaround by making a new variable to store it until it's time to rejoin the string. - that's to solve tomorrow!

Activity

Write a comment...

Thursday 26.10

APP IDEA & USER STORIES

My IDEA is to create an app that helps an adult language learner at beginner level to learn basic vocabulary.

User Stories

Design #1: from stories

Design #1: from User Stories

App Map

+ Add a card

To Do...

Overview

TIME REVIEW: Deploy APP with current features to understand this process, before adding more.

- * Complete Basic App features only (random noun generator later)
- Complete OOP/Class for other vocab lists, now this one is working
- Complete error handling and error testing for current program - this should help with data glitches in csv too
- Explore and practice app deployment to be confident with how this goes and how much time it takes. - more testing here.

App Features

Menu

Quiz feature

Random Noun Generator (relegated for now)

Storage/Backup Tasks

Documentation Tasks

+ Add a card

Do...

Create error handle/condition to prevent sentence punctuation resulting in a wrong answer

Create and error handle/condition to eliminate extra information resulting in a wrong answer or reconsider option of including extra information in CSV.

When happy with these, to next task.

Able to eliminate punctuation and articles from function, but even then conjugation variations meaning matches are skipped. I could account for this with lemmatization I was reading about and find a package to map this and account for it. But I think this is getting beyond the scope of my app. I'm going to make the quiz very simple and just move on to the classes.

Not making much progress. Exhausted and unable to problems solve. Still trying to fix simple quiz.

Brain is fried today - I've made some functions and classes I can use overall. I haven't moved on to bash. I spent a lot of time debugging my code, but probably also couldn't let go trying to troubleshoot extra steps that I could cut to make the bare min and get through.

Add an item

Activity

Write a comment...

Anna Cox just now

Calling it a day now as still unproductive and poss making more problems

Have done:

- random list generator
- function to generate list for all topics
- list cycle feature
- function to generate list cycle for all topics
- vocab quiz feature
- function to generate quiz for all topics
(this was the hardest to let go of the extras on)
- set up class for list generator as needed for all other features, but having trouble getting program to run smoothly between files, so that's the starting point tomorrow.
- glad I wrote the list, as though very slow and not very productive, still some progress was made.

Hide Details

⋮ Edit · Delete

Friday 27.10

APP IDEA & USER STORIES

My IDEA is to create an app that helps an adult language learner at beginner level to learn basic vocabulary.

User Stories

Design #1: from stories

Design #1: from User Stories

App Map

+ Add a card

To Do...

Overview

TIME REVIEW: Deploy APP with current features to understand this process, before adding more.

- * Complete Basic App features only (random noun generator later)
- Complete OOP/Class for other vocab lists, now this one is working
- Complete error handling and error testing for current program - this should help with data glitches in csv too
- Explore and practice app deployment to be confident with how this goes and how much time it takes. - more testing here.

App Features

Menu

Quiz feature

Random Noun Generator (relegated for now)

Storage/Backup Tasks

Documentation Tasks

+ Add a card

Do...

work on errors and testing

create a bash script

Have realised my git was NOT set up properly! Recreate path!

Refactor classes/objects created to run more tidily from main.

Friday's Doing List

70%

Address Git problem and check all is being added on remote

Double check error messages - possibly caused by gitignore?

Refactor program flow from main; streamline class and instances

I ended up having to trouble shoot my features today to get them running smoothly from main with any input - happy with it so far, looking much leaner

get some error handling factored in

create a bash to run program; as is so far, from file;

I'm unsure how bash is going to work with main prompts to run... this is going to influence how I finally set up main

Get all features running from main; from wrapper prompt

Get all features running with new-to and from main menu

Having a problem with random generator on second cycle - iron out with error handling, as repeats welcome in this program

Power-Ups

+ Add Power-Ups

Actions

→ Move

Copy

Make template

Archive

Share

Activity

Write a comment...

Anna Cox a few seconds ago

Am pretty happy with progress, though slow. Much relieved bash ran and program ran with it no worries (so far), some refactoring and debugging today.

Tomorrow error handling, color and documentation finishes are next steps on the board.

Then I can look at time for any extra flourishes if everything is running well incl definition check and random noun search.

Bash was a big hurdle for me, very slow not knowing how it will go/how to apply it to my code. It turned out I didn't break it!

⋮ Edit · Delete

Saturday 28.10

Daily Chunks SATURDAY 28/10

in list Doing Now

Notifications: Watching

Description: SATURDAY's To Do Chunks

- Add packages for formatting vocab cards, pretty tables and color
- Integrate CLI package? Is this possible?
- Dictionary packages.
- Error handling

Vocab Cycle Feature: Flashcard Formatting

Hide checked items Delete

100% Pretty Tables-to-organise and format info

Add an item

List formatting - color

Hide checked items Delete

100% Add Color-to-distinguish german/english-target-vocab-items

Add an item

Activity

Write a comment...

Anna Cox 3 few seconds ago

I got the packages in today, had to refactor most of my program as the table changed how I got the variables for rest of main.
Much less confident I'll be able to finish the task in time.

[Edit](#) [Delete](#)

Sunday 29.10

Daily Chunks SUNDAY 29/10

in list Doing Now

Notifications: Watch

Description: SUNDAY's To Do Chunks

- Finish error handling and testing.
- Even though due today I'm still too slow finishing- I should be able to finish just keep going through the steps and get it done.
- Today, get through testing and then go from there.

Error Handling

Hide checked items Delete

100% Most error-handling done for features – some delay-trouble shooting new-structure in refactoring and composing:
main-new-structured – became apparent-required-in-error handling/testing – code-working okay-but testing and error handling-made-structural problems apparent:

testing-for-main-working-tests-for-features-need-completing

Features-spliced-out-into-own-files accordingly; without-breaking list:

Add an item

Activity

Write a comment...

Anna Cox completed Features spliced out into own files accordingly, without breaking list, on this card
just now

Anna Cox completed testing for main working. tests for features need

[Edit](#) [Delete](#)

After Sunday, I am late, I just keep going.