Tomislav Simic

Professor: D. Marrero

Class: 12P

Date: 2024-02-29

Name of the application: FurBabyBoarding

Overview

Keep track of available boarding spots based on actual events vs daily spot allocation. No real pet or owner information is gathered due to HIPPA-like vet patient data protection.

Reason

The current application in ‘use’ is being avoided as the tracking is based on static date entries, with disregard to actual occupancy at any given time. Current workaround implemented is sticky notes that don’t have a proper tracking and convoluted process that is difficult to train new hires on.

Target Audience

* Staff working with animals:
  + Young man and women, in their 20’s with completed high school and some college.
* Animal hospital management (if the application is found feasible):
  + Different age and gender doctors and managers

Objectives:

*Prompt 1*

Allow users to enter pet name and assign a specific type of kennel. This step has been completed.

*Prompt 2*

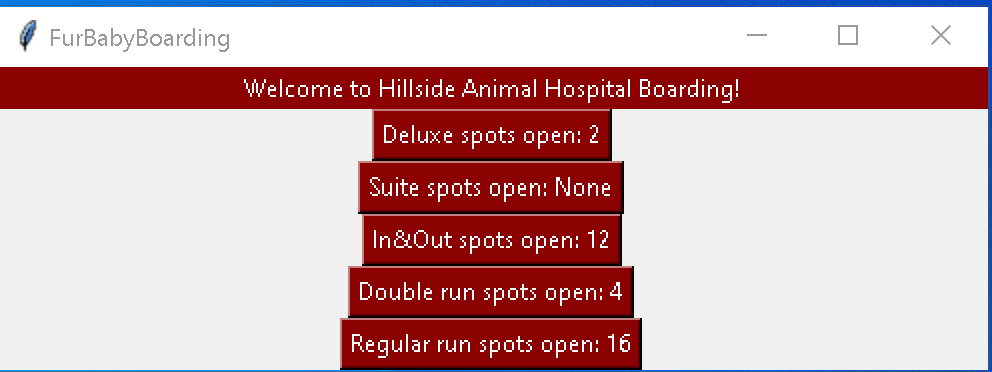
Calculate open spots based on live data and display all available spot and their quantities with 2-week look ahead.

*Prompt 3*

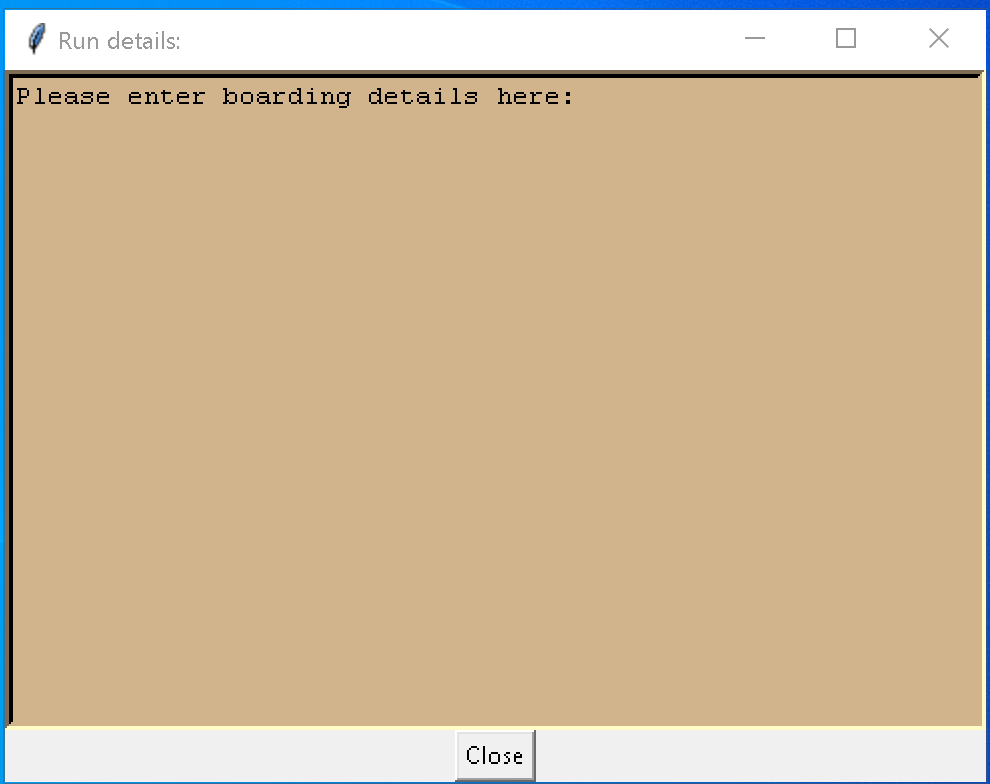
Display a warning if attempting to book a spot that has been already filled. Otherwise, prompt for user to enter basic pet information

Sample Wire Screen

*Window 1*



*Window 2*



Types of kennels and their quantities:

* 2 -3 pets:
  + Deluxe = 2
  + Suites = 7
  + In/Out :
    - Front = 7
    - Back = 5
  + Regular = 16
* Single pet:
  + Cages:
    - Large = 2
    - Medium = 4
    - Cat = 7
    - Rabit = 2
  + Stalls = 2

Current State:

I still have to create functions to calculate current boarders vs incoming, which will react with either providing the input window or show warning and ask to proceed.

Still need to create a function to handle database entries.

I’m open to suggestions if anyone knows a better way to put the classes together instead of random class for the 2nd pop-up window but anything I have tried so far, either messes up rest of the code or is littered with errors and PEP8 violations.

Updates made until 3-5-2024:

Added functions:

* Fielded -> Creates set of fields to represent available runs instead of showing them within the button.Updates made today:

Added classes:

* User input verification test unit -> checks user’s input against correct format and shows corresponding messages, and returns tuple of date segments

Updates made 3-7-2024:

Added functions:

* Reaction -> Implements changes to the display field based on logic and tested it
* Spots\_available -> Checks the suites in dictionary for availability and updates the information based on user’s selection

Additional changes:

* Moved the READme.txt file to new folder (FurryResources)
* Modified code to reflect the READme.txt location
* Added picture for testing as additional label for the requirements
* Added icon to all the windows
* Made the App class UserInput’s class subclass
* Added label to entry window for user’s visibility
* Collapsed 2 functions for help window into a single get\_help function
* Added another label to show a picture