Freedge Tracker README

Authors:
Ellie Kobak, Kayln Koyanagi, Liza Richards, Madison Werries, Ginni Gallagher

Date created:
2/8/2022

Description:
This Freedge notification system is a prototype of a system intended for Freedge Organizers to easily and efficiently update the activity status of the community fridges involved in their organization. The organizer will be able to interact with a window at system startup where they will have the option to send notifications regarding whether or not a freedge is currently active, when a freedge was last notified regarding their active status and view a list of all freedges in the organizations and the contact information for their caretaker.

The main intention of this system is to organize all of Freedge's data in a database that allows easy access to information on each freedge involved in the organization. Unless a Freedge Organizer manually sends a notification to a fridge through the main interface, each fridge that is currently active will automatically be notified after 90 days. This notification will prompt the fridge caretaker if the fridge is active and request a 'Still active' or 'No longer active' response. The responses are recorded and updated in the database. These functionalities are all designed to make the job of the Freedge Organizers easier and to increase organization within Freedge.

Reason for Project:
This project was created for Professor Anthony Hornof's CIS 422 class during Winter of 2022 at the University of Oregon. It's purpose is to help Freedge Organizers quickly and easily update the activity status of freedges in their database so that users will be able to know which fridges to visit and add food to.

How to Compile and Run Program:

1. Navigate to the Freedge Tracker github repository: https://github.com/kalyn-k/Freedge-Tracker
2. Download this zip file "Freedge-Tracker.zip" to your computer
3. Open the terminal application using "Command + space bar" to open the search window and type in "terminal"
4. In the terminal window, change the directory to where the Freedge Tracker system is stored. In most cases, this command would be "cd Downloads/Freedge-Tracker-main". Otherwise you can use "cd *your file path*/Freedge-Tracker-main and hit enter.
5. If the contents are in a .zip file, you can unzip the file using "tar -xzvf Freedge-Tracker-main.zip" into the terminal command line, or you may navigate to where the .zip file is stored on your computer and double click or right click on the file.
6. Once in the proper directory, run the software by typing "python3 freedge_tracker.py + enter"
7. The system is now ready for use, select the "Create Database" or "Load Database" buttons on the main menu window to upload your data and begin!

Additional Setup:
For this program to run as intended, the file that holds all of the freedge data must be in a .csv comma separated format. The first line of the file must contain a comma separated header of the format:

<project_name> <network_name> <street_address> <city> <state/province> <zip_code> <country> <main_contact_method> <date_installed> <contact_name> <activity_status> <phone_number> <email_address>

The rows that follow the header will contain the corresponding freedge information in the same order shown by the header. Additionally, each installation date and the date of the last status update have to be in the formation:

YYYY-MM-DD

Every header under the constants file NEEDS to exist in the CSV file with the same name. The user can change the names of headers in the InternalData folder in the “freedge_constants.py” file to match the format of the CSV file. NOTE: No fields in the constants file can be deleted or missing from either the freedge_constants.py or the CSV file.

Software Dependencies: Python 3

Directory Structure:

1. “AdminInterface” includes the files that enable the whole system to run, including all of the tkinter display file set-ups.
2. “FreedgeDatabase” includes all of the components that work together to create the freedge database.
3. “InternalData” contains a constant file and a path file that allows easy updating of global variables.
4. “NotificationSystem” includes all of the files that work together to send the user notifications.
5. “Test_data” is a folder that contains all of the test csv files that can be uploaded into the system. It also includes one database file that contains out of date fridges for testing.
6. “Documentation” is a file containing the system requirements, system standards
7. The “Freedge-Tracker.zip” is a zip file containing all the files necessary to run the system on your local machine with the instructions above.
8. The “freedge_tracker.py” file is the main driver for the system. This file will be run in the terminal to start the software.