Criterion B: Design

The product will be developed in python and run on a linux computer dedicated to the task. The client will provide an account to that computer to develop and run the program. Two threads will be used. One to interact with the user and the other to monitor the powerwall and shutdown/bring up devices.

Textual User Interface

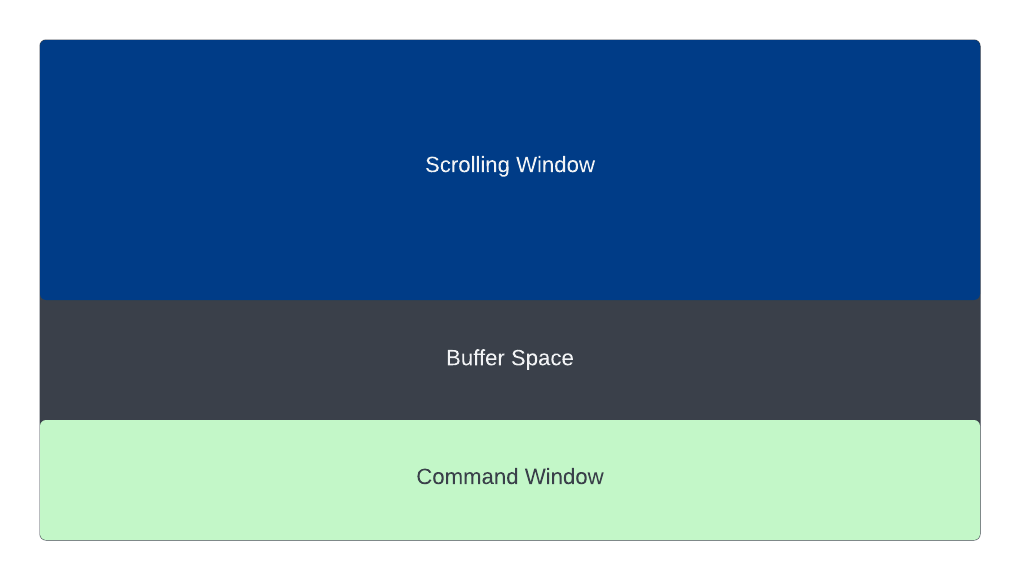
[](https://lucid.app/documents/edit/9f3c4158-5fef-4c3b-b957-453ee678cc18/0?callback=close&name=docs&callback_type=back&v=312&s=510)

Figure B.1: Textual Interface Using *curses*

Functionality of each window

* Scrolling window - displays grid status and power use every minute and devices being shut down and brought up
* Command window -displays three options 1. Test shutdown, 2. Test bring up and 3. Exit program and information about the powerwall (name, charge percentage and capacity)
* Buffer Space - prevents interference of libraries printing error messages without using curses

Main thread (command thread)

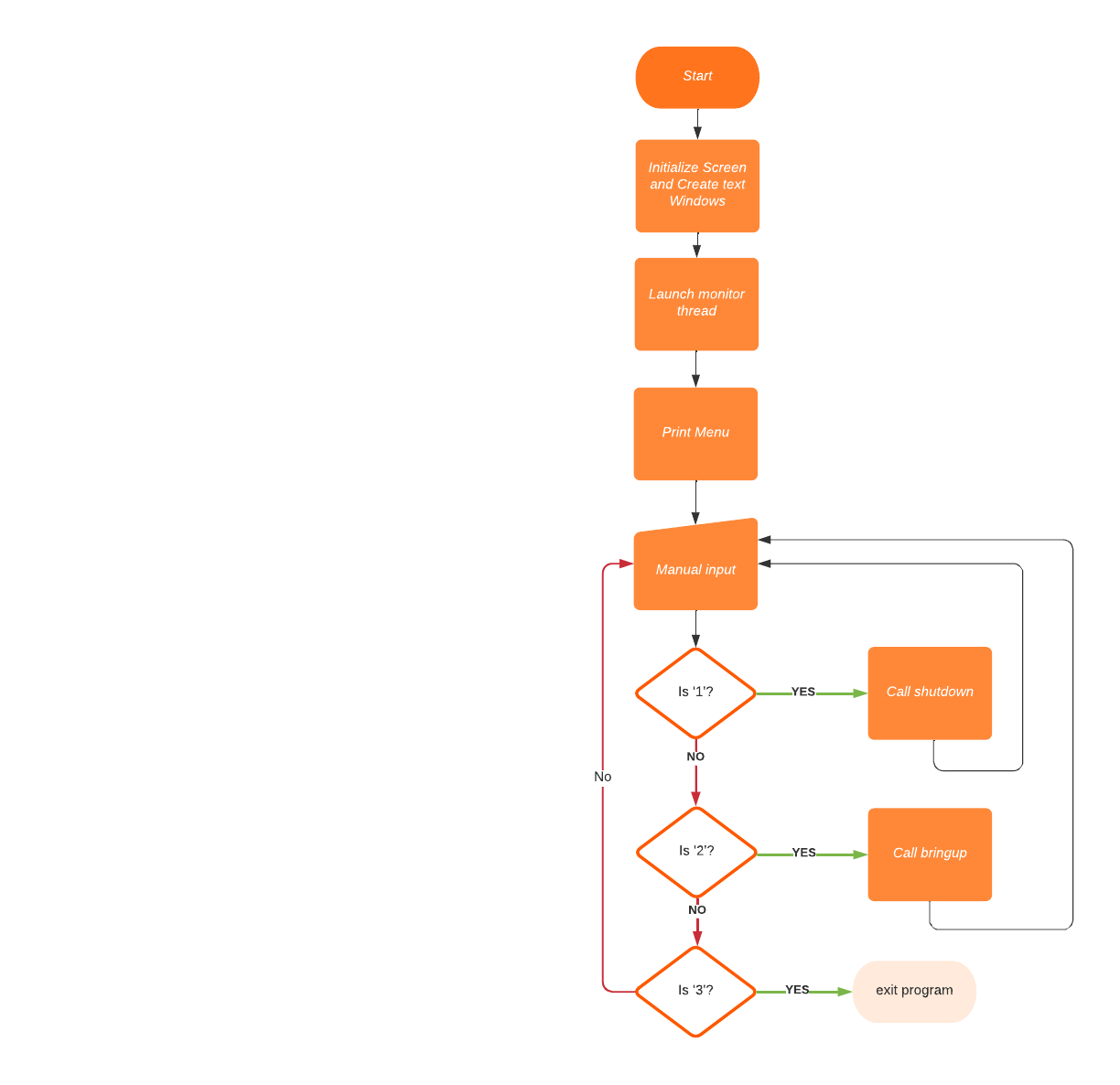
[](https://lucid.app/documents/edit/2b3b1104-2ef4-47ce-8d8f-d374411b6e24/0?callback=close&name=docs&callback_type=back&v=727&s=612)

Figure B.2: Main Thread Flowchart

Monitor Thread

This is the high level flowchart of the monitor thread.

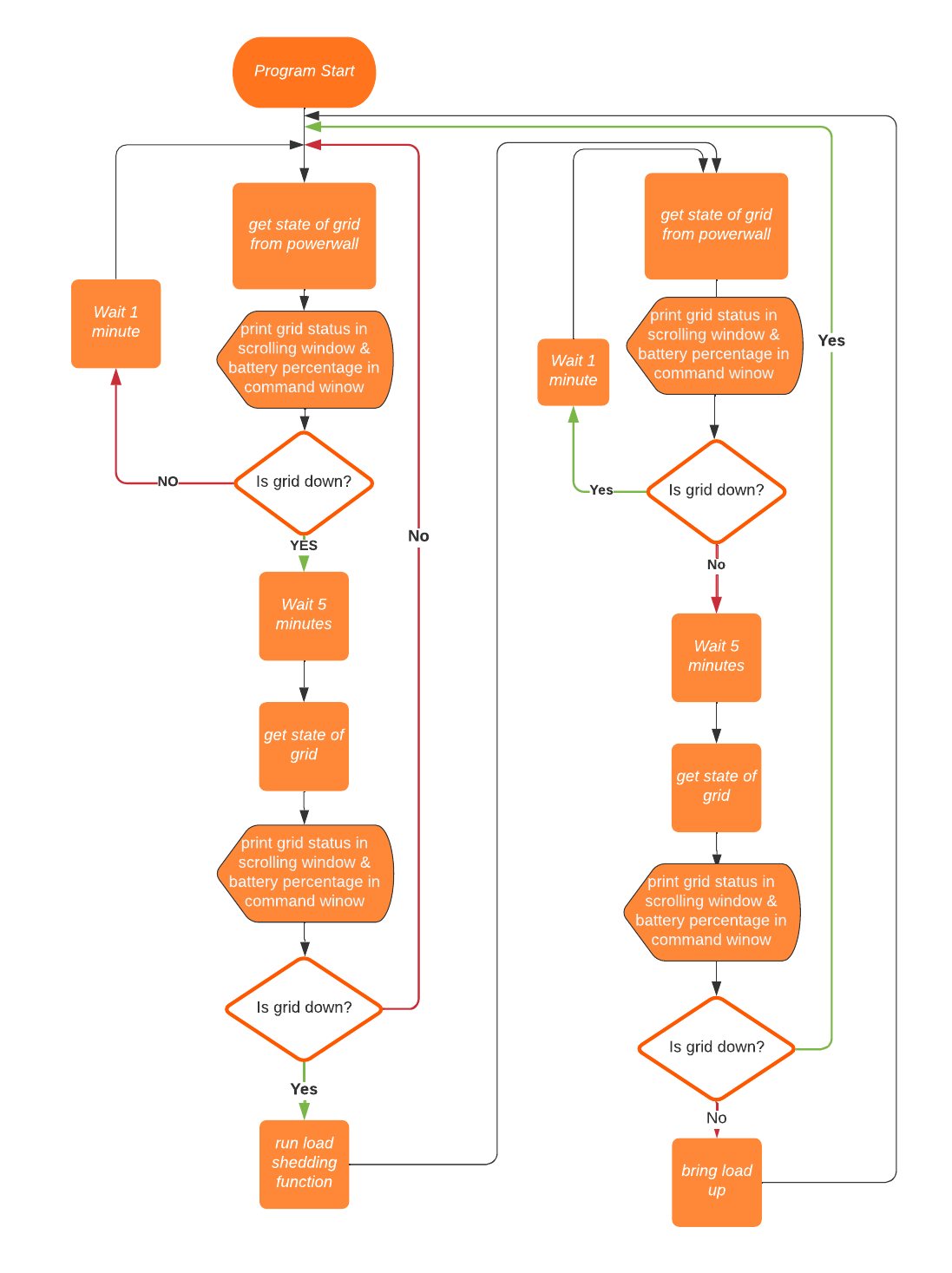
[](https://lucid.app/documents/edit/01b750bb-f668-497a-83f4-b684d005dead/0?callback=close&name=docs&callback_type=back&v=1161&s=548)

Figure B.3: Monitor Thread Flowchart

Powerwall Access

The client has a tesla powerwall installed. In order to know if the grid is down the power wall will be probed using the *tesla\_powerwall* python module.

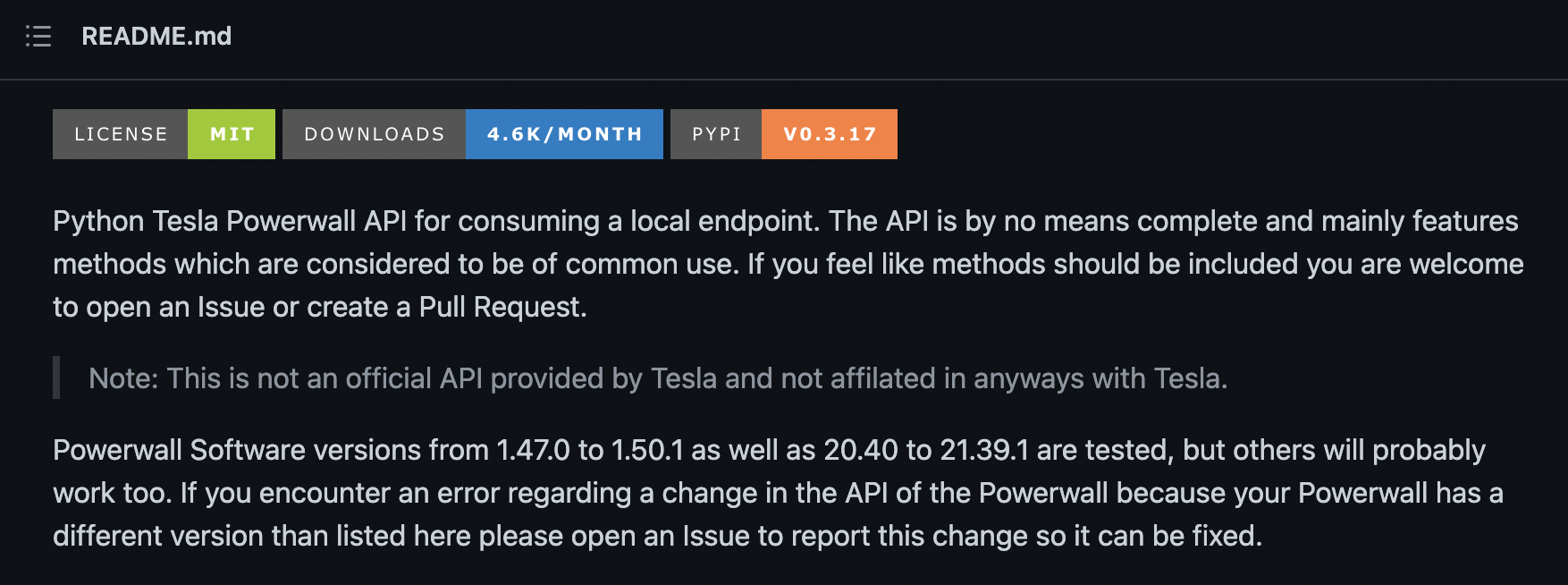


Figure B.4: *tesla\_powerwall* Python Module Description

Load Shedding

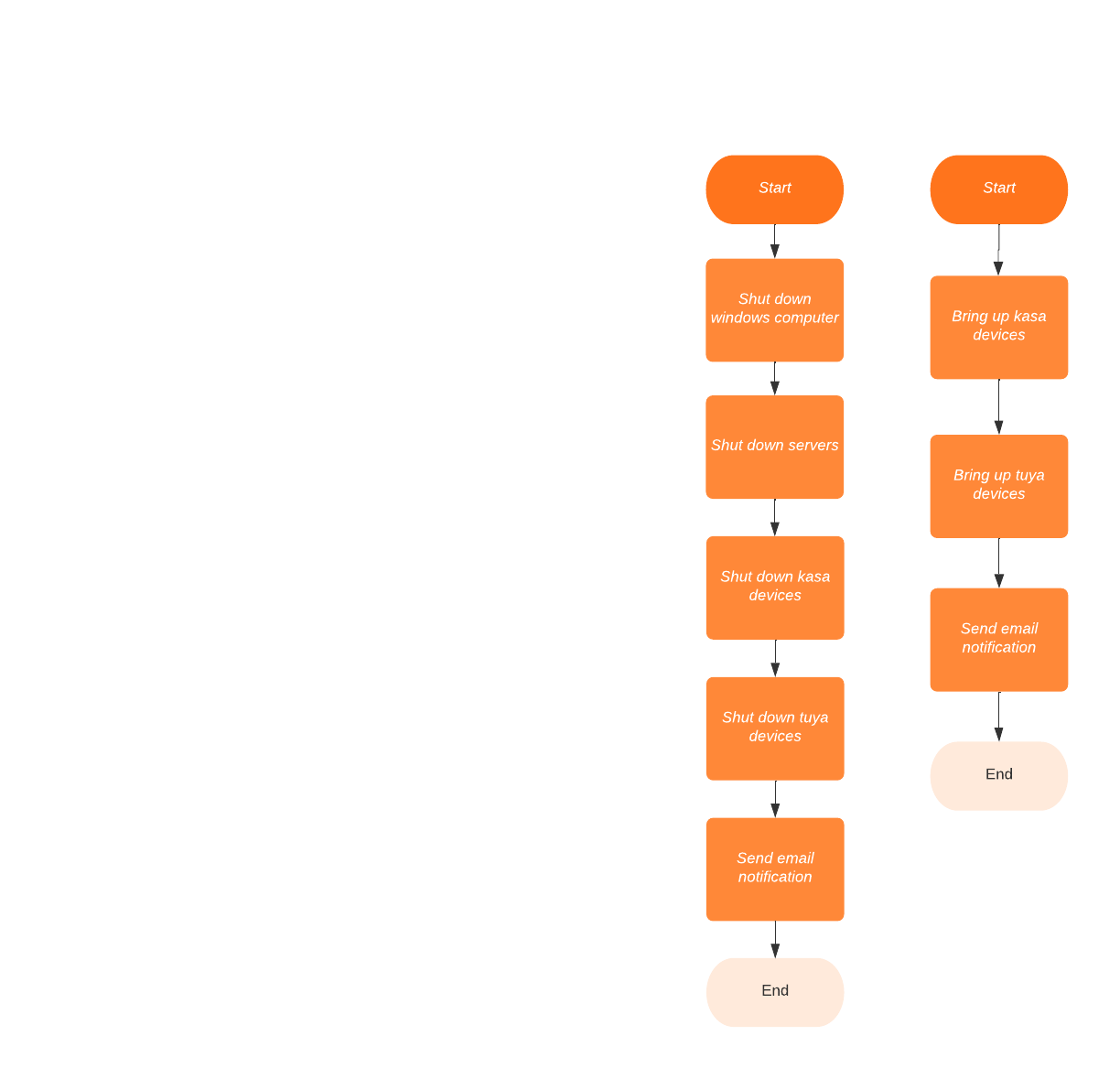
[](https://lucid.app/documents/edit/6b62d96c-68b7-4ed7-866f-9f729422cd25/0?callback=close&name=docs&callback_type=back&v=440&s=612)

Figure B.5: Main *shutdown* and *bringup* Functions

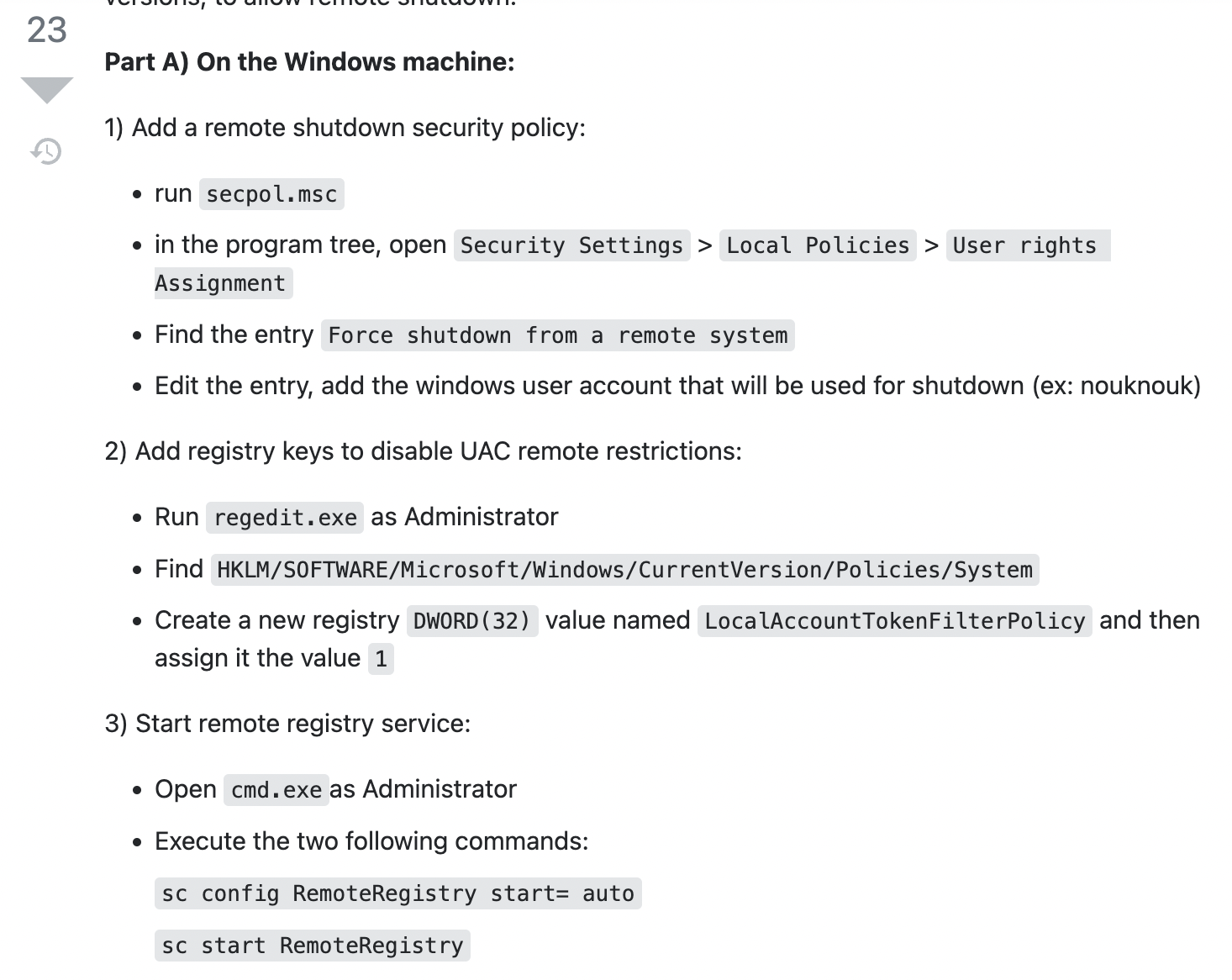
The load shedding function will perform the following tasks:

* Shutdown windows computers
* Shutdown Unix servers
* Turn off Tuya Smart plugs, switches and bulbs
* Turn off the tp-link Kasa Smart plugs

In order to store the required information for all computers there will be two files (*windows.txt* & *servers.txt*) that the user can edit.

Windows Desktops

Windows computers must be prepared to be able to be shut shutdown remotely.



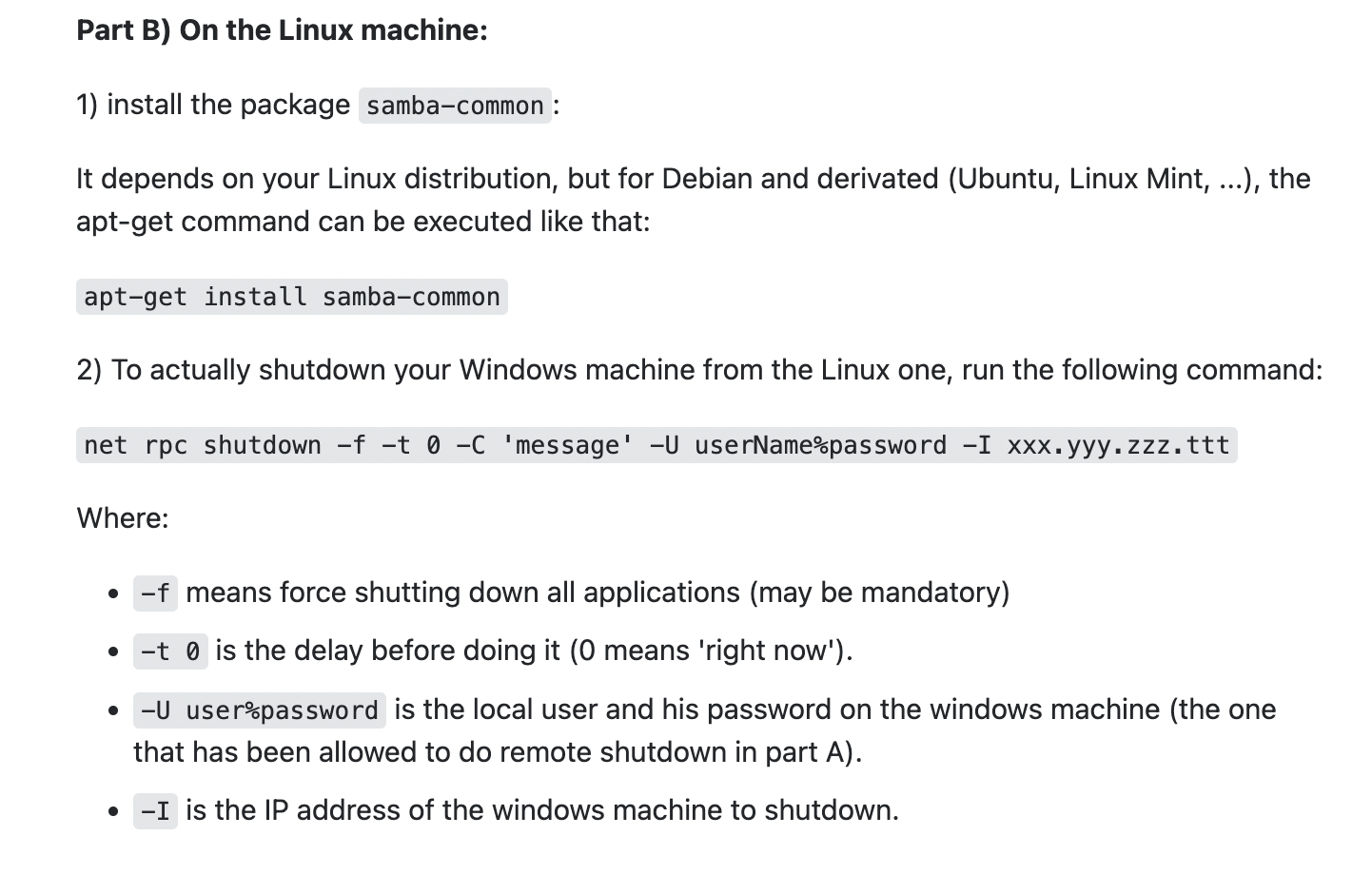


Figure B.6: Windows Computer Setup Directions

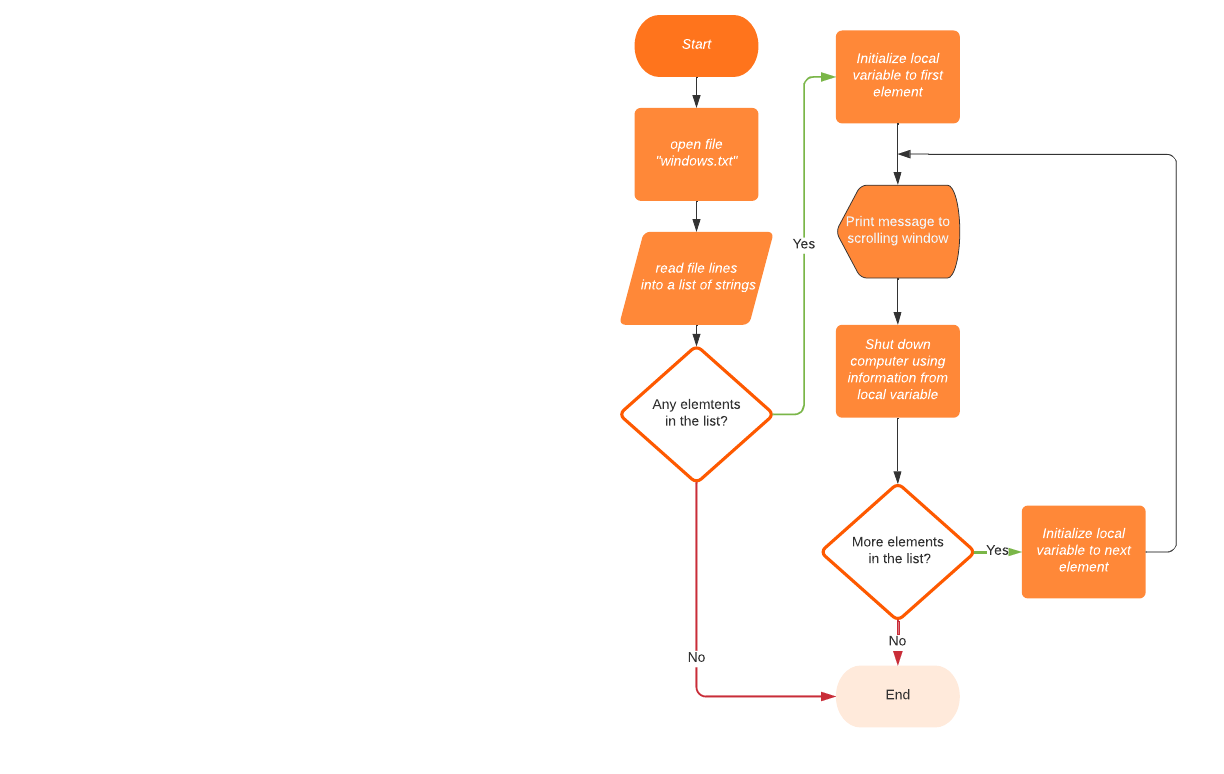
[](https://lucid.app/documents/edit/5cf5f6e8-5622-4180-9935-5f53884d79b6/0?callback=close&name=docs&callback_type=back&v=831&s=612)

Figure B.6: Function to Shutdown Windows Desktops



Figure B.6: windows.txt Format

Servers

In order to shutdown linux and solaris servers *ssh* will be used to execute the *shutdown* command remotely. Linux computers can be shut down as follows:

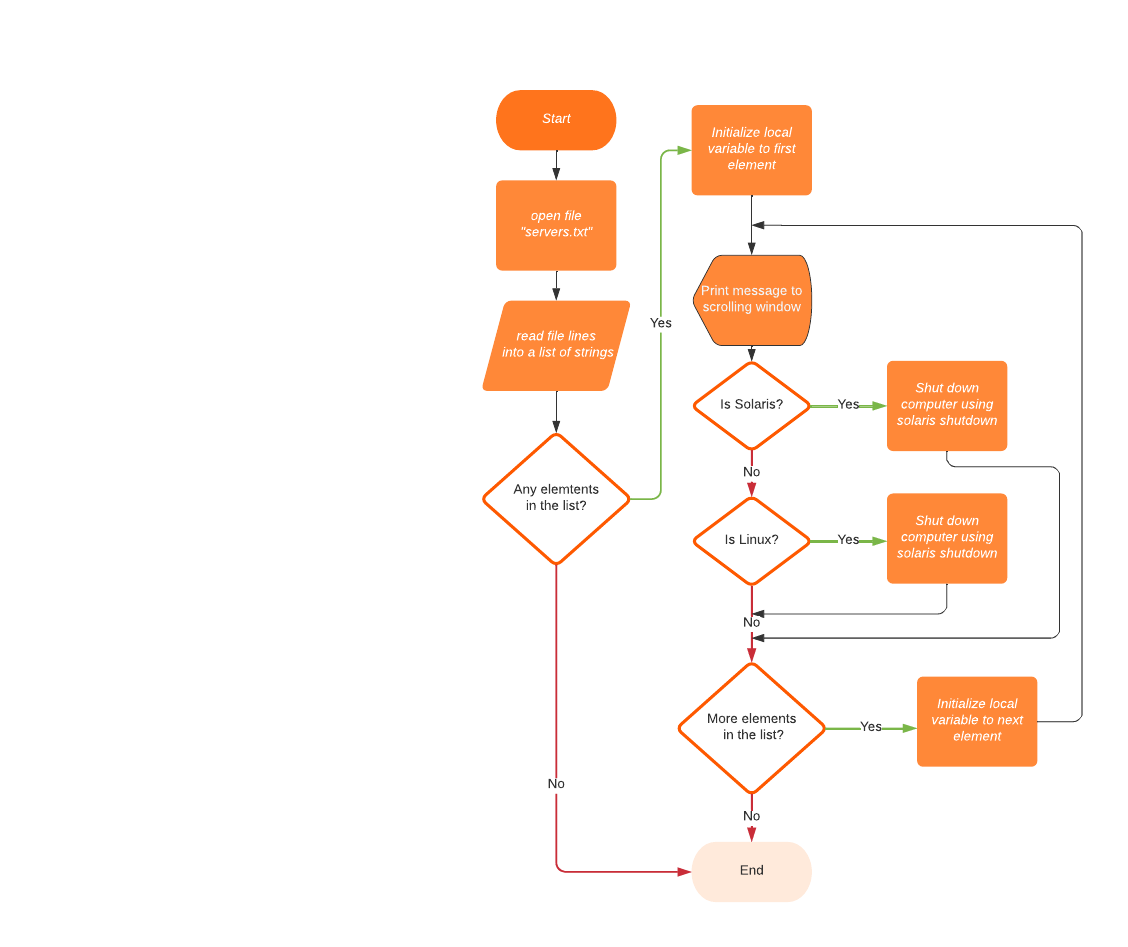
[](https://lucid.app/documents/edit/bdcd4b3d-54bf-4eb8-9483-c7edc4e85dcf/0?callback=close&name=docs&callback_type=back&v=420&s=564)

Figure B.7: Function to Shutdown Servers

To avoid storing usernames and passwords for linux and solaris servers *ssh-copy-id* command will be used to copy account credentials to all the servers that will be shut down.



Figure B.6: servers.txt Format

Kasa Devices

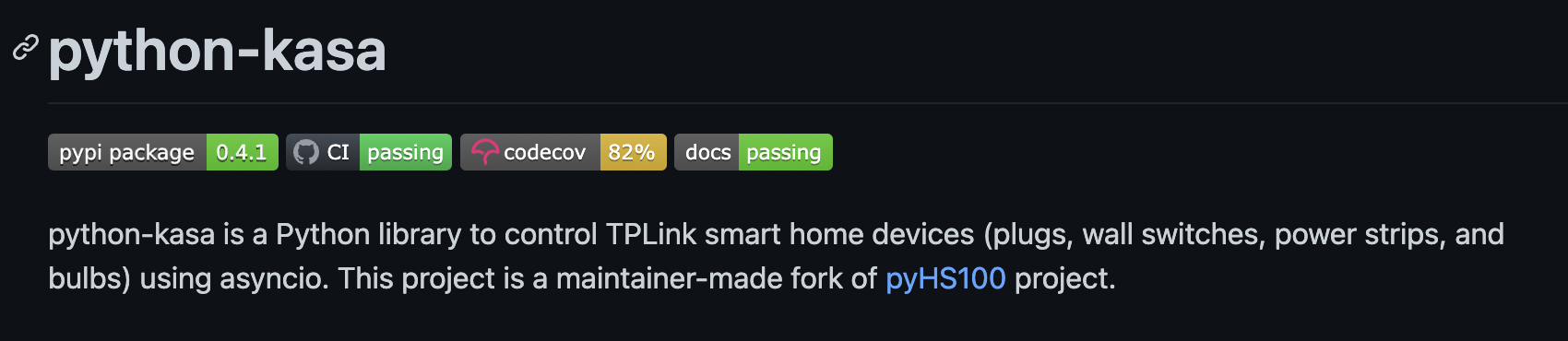


Figure B.6: Module for Controlling Kasa Devices

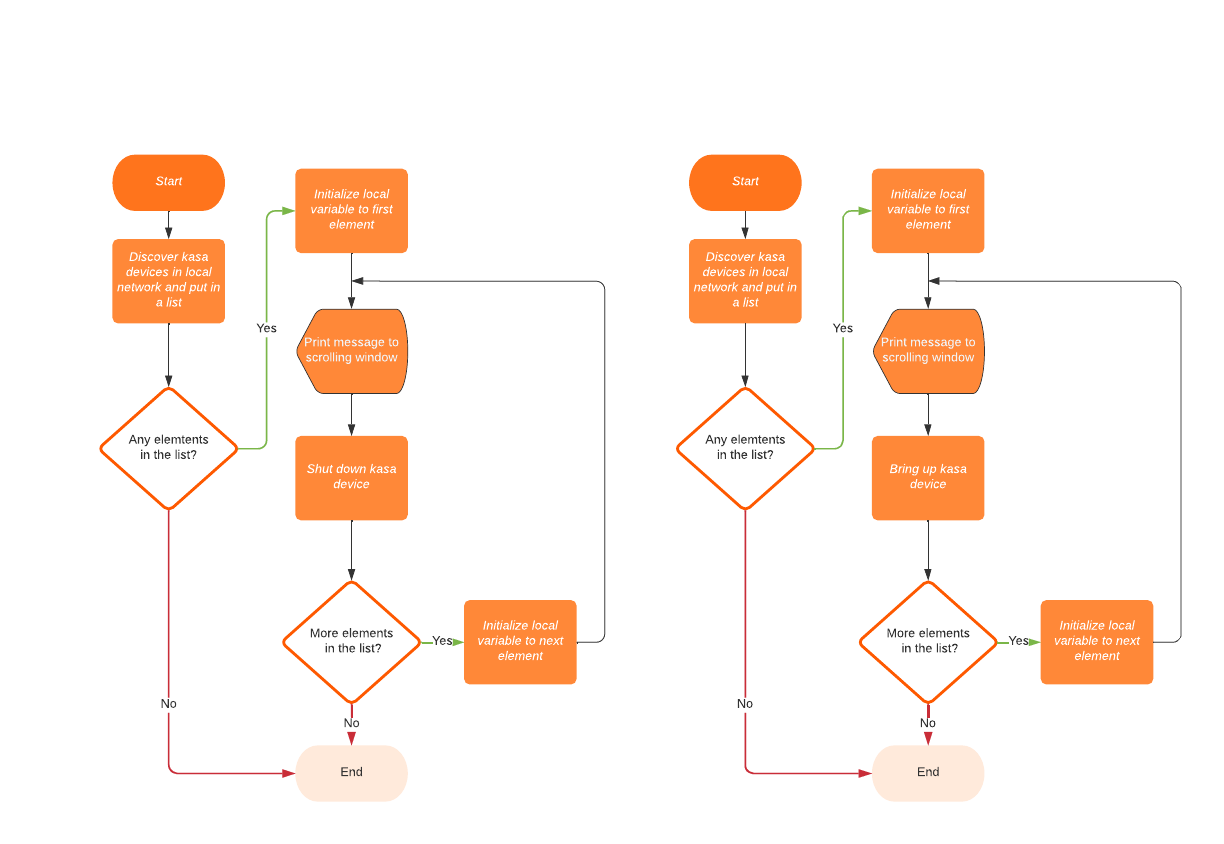
[](https://lucid.app/documents/edit/23418db5-4d54-441c-925e-be0f0414a6db/0?callback=close&name=docs&callback_type=back&v=434&s=612)

Figure B.7: Functions to Shutdown and Bringup Kasa Devices

Tuya Devices

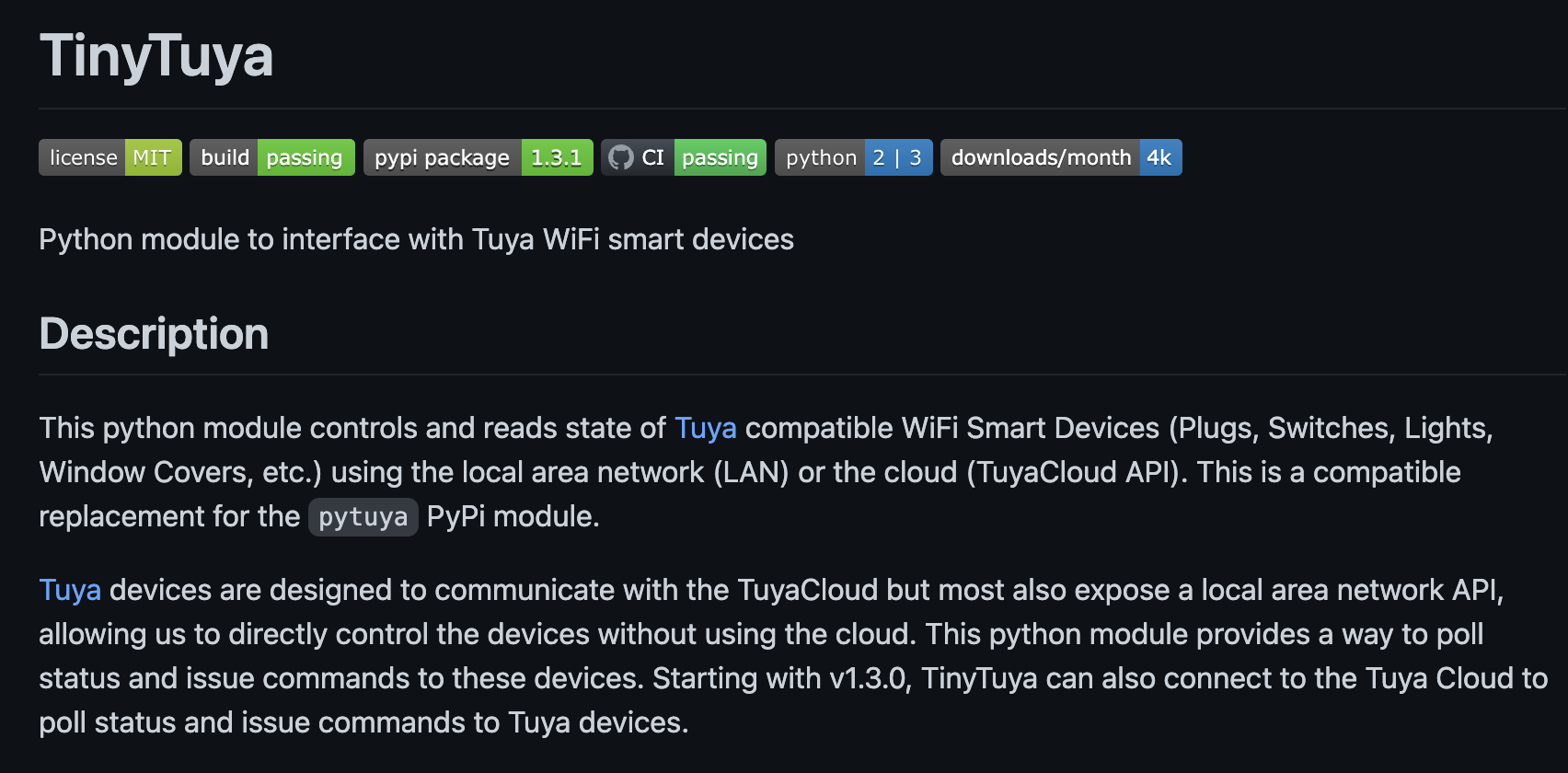


Figure B.6: Module for Controlling Tuya Devices

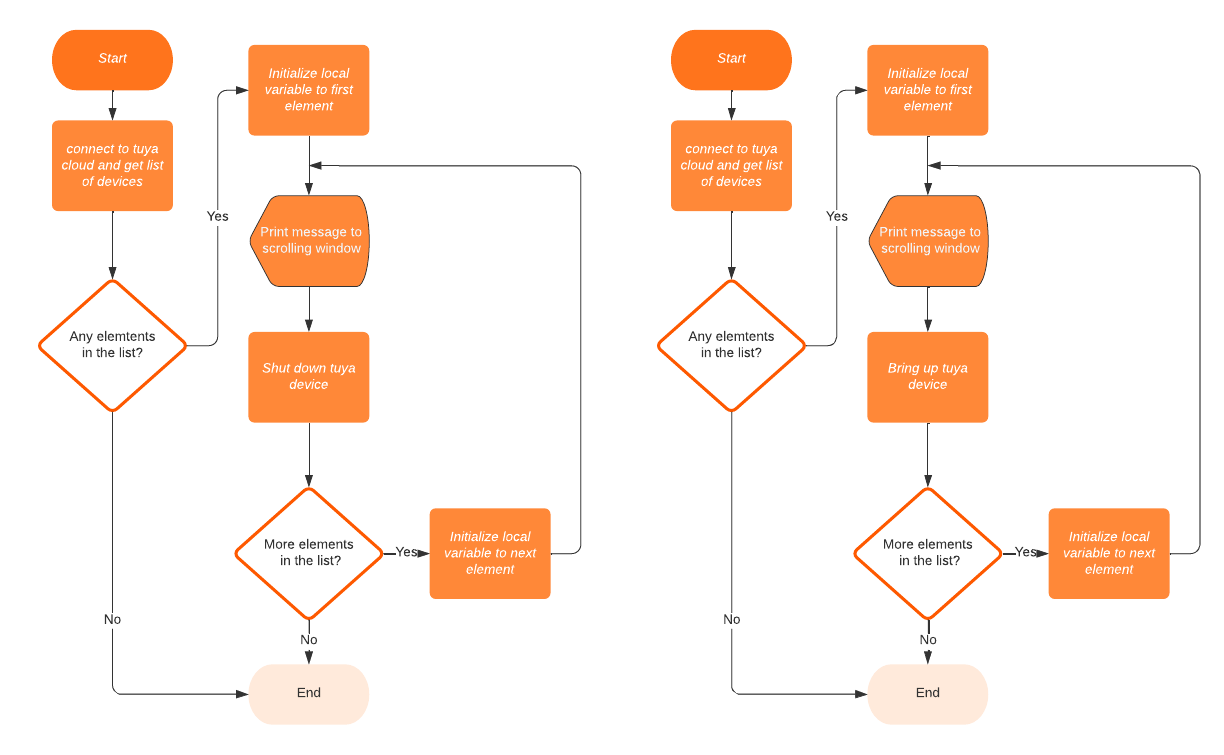
[](https://lucid.app/documents/edit/99b99d7b-a23a-4b98-b8ff-014e181ff919/0?callback=close&name=docs&callback_type=back&v=258&s=612)

Figure B.7: Functions to Shutdown and Bringup Tuya Devices

Test Plans

Test Plan Main loop:

* Manually disconnect from the grid
* Confirm whether the disconnection is detected by the program and whether it runs load shedding function
* Manually reconnect to the grid
* Confirm whether the reconnection is detected by the program and whether runs load bringup function

Test plan Windows desktops:

* Make sure computers on
* Prepare the computers as indicated in the [instructions](https://stackoverflow.com/questions/9936986/shutdown-windows-machine-from-linux-terminal)
* Run a test from the machine where the program will be running
* Determine whether the windows desktops were shut off when disconnected from the grid

Server Test Plan

* Make sure servers are on
* Prepare the servers to ensure ssh commands can be used without username and password
* Run a test from the machine where the program will be running
* Determine whether the servers were shut off when disconnected from the grid

Tuya and Kasa test plan

* Turn on all Tuya devices
* Determine whether are devices are indemnified with the scan function
* Run a test from the machine where the program will be running
* Determine whether the devices were shut off when disconnected from the grid
* Determine whether the devices were brought up when reconnected to the grid

User interface test plan

* Determine whether pressing 1 starts a test shutdown when grid is still connected
* Determine whether pressing 2 starts a test bringup when grid is still connected
* Determine whether pressing 3 immediately exits the program and restores the default terminal