**AutoBeetle (*working title*) – Automated Beetle Monitoring**

Becky Heath, Ellie Bladon, Willow Dalehite (March 2024)

AutoBeetle of autonomous captive insect monitoring… *general introduction, contributors, funders etc.*

Kit List:

* Raspberry Pi 4B+ (8GB??)
* 256GB Micro SD
* (Infrared) USB cameras
* 15W 5V/3A powering cable for the Pi
* **Powered???** USB hub
* ????
* MicroSD adaptor

For setup:

* Screen (USBC or HDMI micro connection)
* Keyboard and mouse (Bluetooth or USB)

Instructions:

**Setting up Raspberry Pi Basics**

1. Download the Raspberry Pi imaging software (this writes the operating system to an SD card). [You can get it here](https://downloads.raspberrypi.org/imager/imager_latest.exe).
2. Insert the SD card to your computer, you’ll probably need to use a hub or an adaptor.
3. Select Raspberry Pi 4, select the recommended OS, and select the SD card.
4. Click “Edit Settings” –

For hostname call it something like AutoBeetle01 (this will be the name of the device on networks). Use different names (e.g. AutoBeetle02) for different devices.

For username and password, keep the username simple and memorable, and write it down. Perhaps use:

Username: bugs

Password: zoology

For Wireless Lan this is access to WiFi – you’ll need to set this up. Eduroam is annoying so I’d maybe just use a hotspot off your phone or something (provided you’re not low on data).

Other setting are self-explanatory.

Then under “Services”, do enable SSH – it’ll mean we can access the Pi remotely (very useful).

**Then hit save, then hit write – this will take a while.**

1. Your operating system is now written to the SD card – woo!
2. Stick the SD card in the Raspberry Pi, connect your screen and mouse etc. then power her up….

**Note:** don’t just unplug the Pi ever. It’s a nightmare to fix so you’ll end up just having to start again from scratch. You can switch it off at the wall, or power off through the interface. Just don’t just yank the cable out.

**Note:** the Raspberry Pi operating system (from Wikipedia) – looks pretty normal, hey?

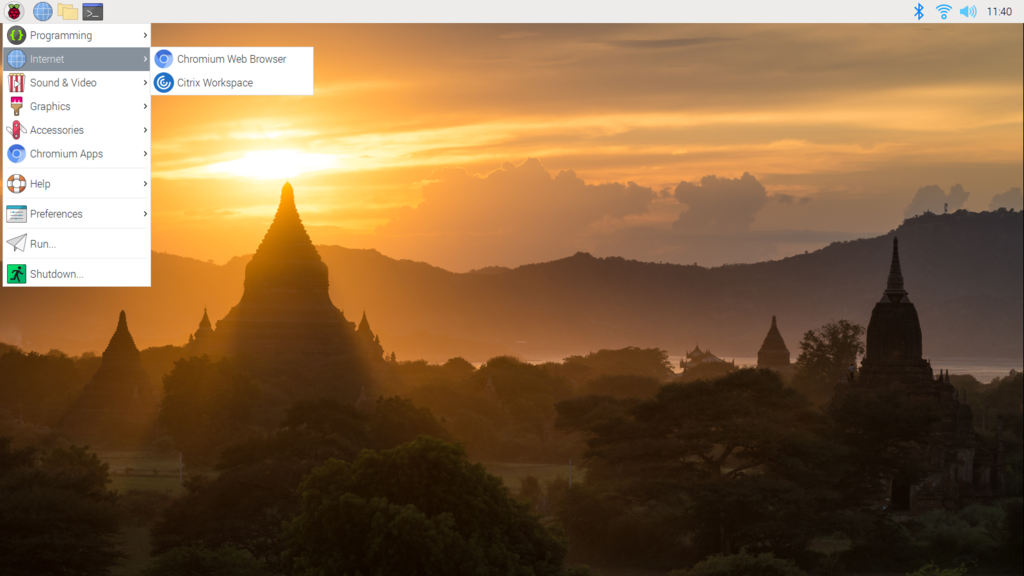
The raspberry is kinda like the windows button where you’ll find all your apps and settings and shutdown.

The globe is the internet.

Files is files.

Then the terminal is the Rpi terminal – which we will be using.

Then the stuff on the right is your Bluetooth, WiFi etc. all pretty self-explanatory.



**Setting up the automated monitoring**

1. Download the required packages by typing the following into the terminal (that’s the black box with a chevron in it on the top taskbar). Press enter after each new line and wait until you can see “successfully installed…” in the outputs. If there is an issue – you’ll need to resolve it. Contact me (Becky) or look up the error online.

“sudo apt update”

“pip3 install os”

“pip3 install time”

“pip3 install opencv-python”

“sudo apt-get install git”

Note: pip is an python package installer. I’m pretty sure it comes with the raspberry pi OS download, but if you get errors like “no such package as pip” or something try running these lines in the terminal:

“sudo apt update”

“sudo apt install python3-pip”

1. Plug the cameras in and check they’ve registered by running the following command in a fresh terminal:

“ls /dev/”

What should be returned is a list of cameras like this:

/dev/video0

/dev/video1

/dev/video3

– **let me know if this doesn’t happen**. We may need to get a powered USB hub – will see.

1. In the terminal, navigate to the desktop.

**“ls”** list directory (gives you a list of files)

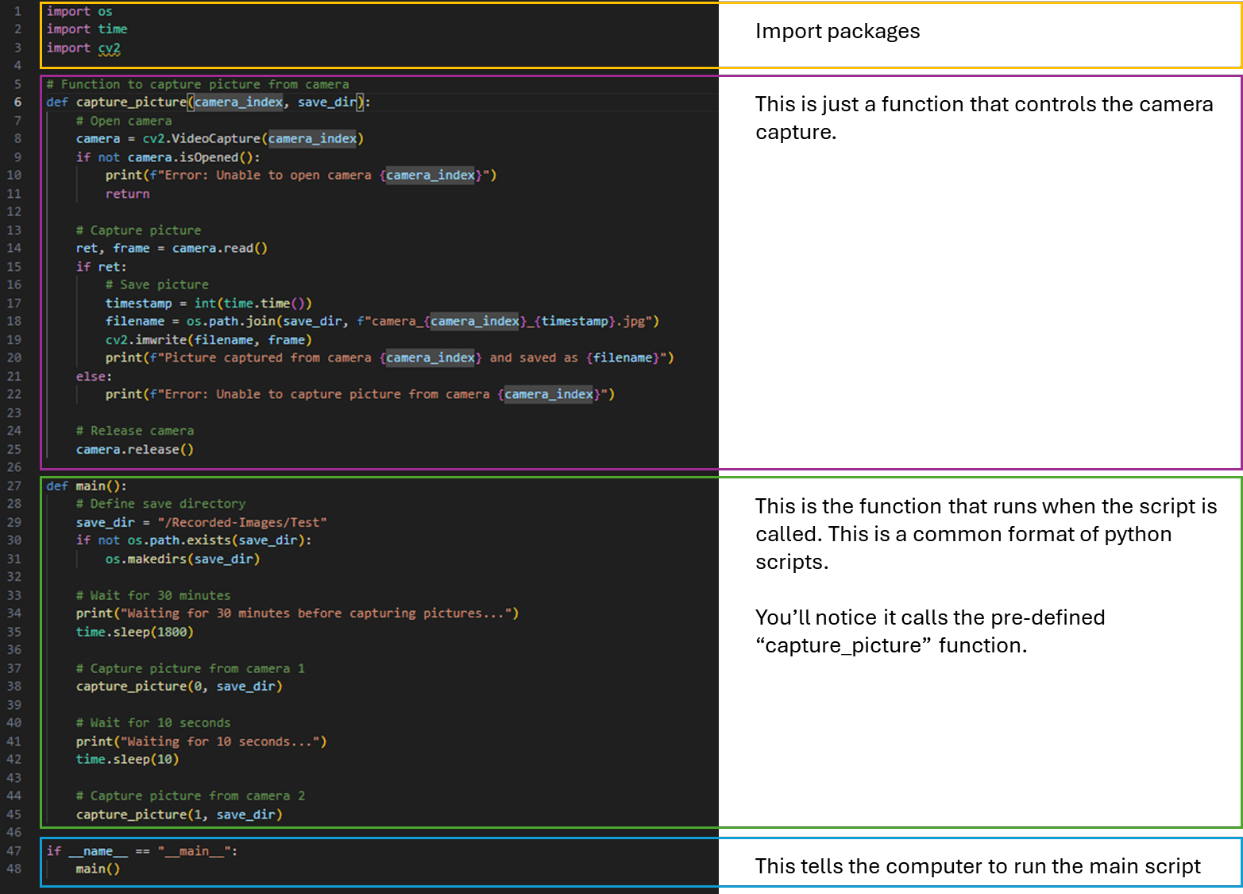
“**cd”** change directory. For example you’d put: “cd home/pi/Desktop” to get the desktop (specifics may vary, sometimes it’s better to go one folder at a time with ls.

1. Clone the GitHub repo (this will copy the code etc. from the repository.

“git clone https://github.com/BeckyHeath/AutoBeetle-BeetleMonitoring.git”

Check out the files. Record.py is the file that’ll run the processes we need.

This is the structure of the python script. Packages are imported first. Then function(s) are defined. Then the processing function “main()” is defined. Then finally the “if name == main…” command is given to tell the computer to run the processing script.



1. Save it and run for a few minutes to see if it’s working.
2. Fix any issues!!
3. Add the script to startup protocols (fill in steps here).
4. Reboot and wait a few minutes.
5. Boot the Pi and check everything is working as expected.