Nothing Ever Happens 'Round Here

0002 – evdev and events for Python

Python Hack Night
December 2018

Agenda

- Welcome
- Python News for the Month
- Introductions
- Monthly Presentation
- Python Jobs
- Upcoming Events

Welcome

- Python Hack Night Goals
  - Build a sustainable, supportive group that values diversity and open practice
  - Promote education and discussion to further people's understanding of Python
  - Collaborate on public projects that are open source
  - Share opportunities and resources to grow the community

Python News for the Month*

- Check out Responder (http://python-responder.org), suggested by Chris
  - Uses modern Python, is asyncio based (better real-time), can be used to write responsive API's or web apps

- Malicious modules found in Python repository (PyPi)
  - Improvements to PyPi are coming, be aware of the modules you are using
    https://www.zdnet.com/article/twelve-malicious-python-libraries-found-and-removed-from-pypi/

- Want to learn Python, check out a quest to learn Python in the open:
  - '100 Days of code - Creating my personal Magical Universe' (http://alpopkes.com/year-archive/)

* Includes interesting stuff from 'Talk Python to Me' and 'Python Bytes' podcasts

Introductions

- Let everybody know who you are
    - Your name
    - Your interest in Python
    - Suggested topics

MONTHLY PRESENTATION

Gamepad Project

- Problem Statement

  People are comfortable with game pad-type controllers but configuring them for general purpose applications can be difficult. Flexibly re-mapping buttons or analog inputs is not easy. Connecting game pad to MQTT would ease integration with community projects like Kegbot.

- Goals
  - Figure out a cool name
  - Collaborate on an open source project
  - Create a game pad controller driver for robots like kegbot
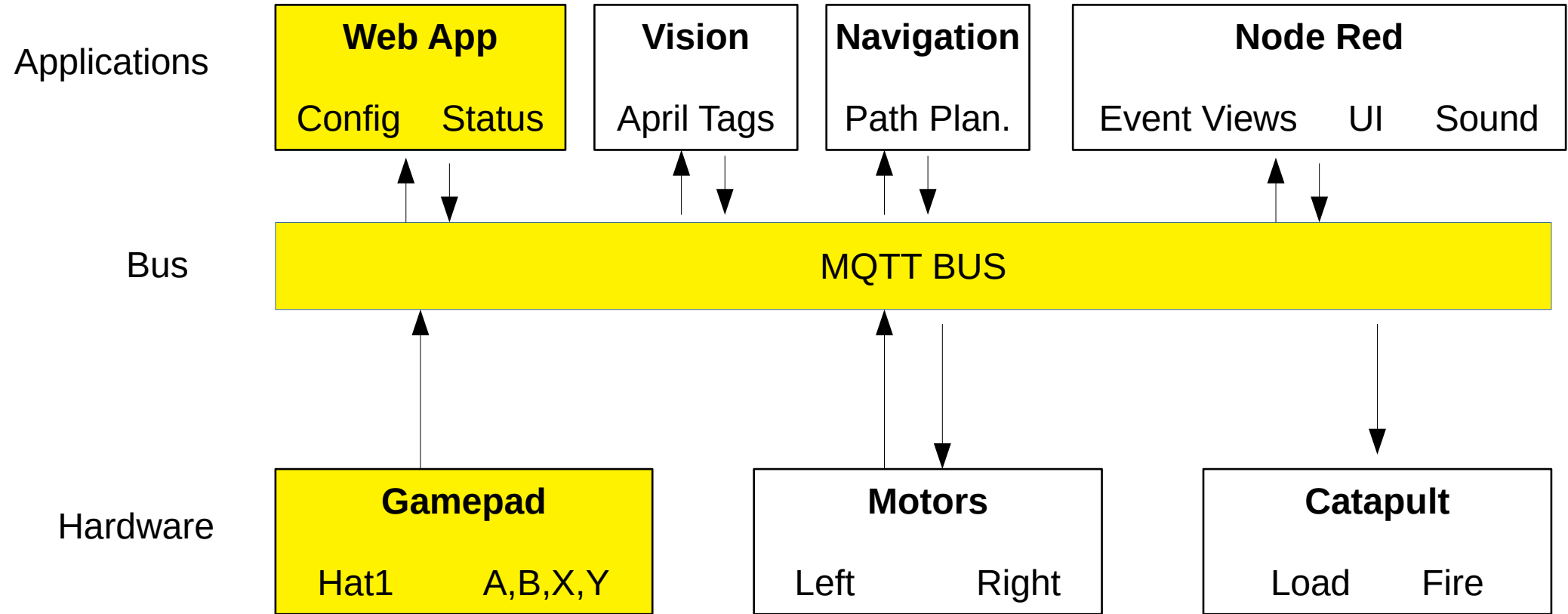 - Get to know Python and some of its popular modules

Features

- Shall support USB game pads
- Shall support set up for game pad by ID, if gamepad is changed
- Shall be able to assign MQTT topics to specified keys/inputs

  – See RetroPie start up for example

- Shall have a diagnostics web page to see what is output from the gamepad

- Could support input keys, if a user had a keyboard connected

- Any other features??

Components

- python 3.x
  - the latest Python ('python 3') on most systems, including RPi, Linux

- evdev
  - a general purpose input interface for Linux (keyboards, gamepads, etc)

- responder
  - for configuration UI. responder is a new async framework for HTTP services (web pages, REST Api's)

- paho-mqtt
  - kegbot's message bus, MQTT is a lightweight pub/sub messaging protocol

- systemd
  - start and control driver script at boot-up or restart.

Architecture

Applications

Web App	Vision	Navigation	Node Red
Web App	**Vision**	**Navigation**	**Node Red**
Config Status	April Tags	Path Plan.	Event Views UI Sound

Bus

MQTT BUS

Hardware

Gamepad	Motors	Catapult
Gamepad	**Motors**	**Catapult**
Hat1 A,B,X,Y	Left Right	Load Fire

evdev Basics

- https://python-evdev.readthedocs.io/en/latest

- *evdev* is a Linux interface that passes events from the computer system (kernel) to your program or script (userspace)

- Some popular uses of *evdev* :
  - Reading data from devices (gampads, keyboards, mice) in /dev/input
  - Injecting events like key presses or gamepad events
  - Reading data synchronously (for event in device.read_loop()…)
  - Reading data asynchronously for more responsive code (asyncio)

Installation

- This only works on Linux systems since evdev depends on the Linux kernel
  - Mac users can a different package like pygame or SDL, which are more portable

- Install Python 3 on a Linux computer (RPi, etc)
  > sudo apt install python3

- Install Python package manager ('pip')
  > sudo apt install python3-pip

- Install evdev module using the PIP for Python3 (not 'pip')
  > sudo pip3 install evdev

Examples

- monitor.sh – monitor selected device for events.
  - Good place to start experimenting with what goes with what
  - Displays event type (EV_ABS, EV_KEY, etc), source (ABS_X, BTN_A, etc) and value (128, etc)

- example1.py – listing devices information

- example2.py – scan devices, map actions and display output.
  - This is a start for autoconfig.
  - Might be better as an asyncio client

Next Steps

- Implement as a module talking to MQTT broker

- Implement with asychio

- Tie a flask instance to MQTT to view gamepad info
  - stick location, button states, etc

- Tie robot drive to gamepad topics on MQTT bus

Hardware Considerations

- Use a general purpose game pad
  - Considered Bluetooth (hard to pair)
  - EasySMX
    - Easy to pair
    - No need for a special adapter
    - Cheap (~$25)
    - Be careful of the mode! (need to have LED1, LED2 lit)
    - Distance is good, but only about 10 feet

  - Need to handle if system is booted w/out controller since it goes to sleep
  - Needed to label the USB adapter since it has no markings at all (!)

Python Jobs

- Senior Python Dev @ Cognaisent
  - AI/Machine Learning
  - http://cognaisent.com/seniorpythondev.html
  - FMI: ted@cognaisent.com

- Indeed
  - Senior Data Engineer @ L. L. Bean
  - Senior Data Engineer @ Vet's First Choice
  - Software Development Engineer 1 @ WEX
  - Applications Engineer @ PTC
  - Software QA Eng II @ Blackhawk Network (Cashstar)
  - Quality Assurance Eng @ ForeFlight

Upcoming Events

- **PyCascade** – February 23-24, Seattle, WA

- **Pycon** – May 1-9 in Cleveland, OH

- **!!Con 2018** – May 12-13 in NYC
  - *The joy, excitement, and surprise of computing*

- **Start Up Maine** – TBD 2019 in Portland, ME

Upcoming Talks

- Planning
  - More Gamepad
  - Basic Python
    - Installing, idioms, etc
  - Jupyter
    - Installation (shared server?)
    - Exploring 'big' data
    - Tips and tricks

- Ideas
  - Data Analytics: Suggestions or interests?
  - Data Science: Numpy, SciPy
  - Web Development. Django, flask, responder, etc
  - Server scripts
  - Robotics: MicroPython, IoT, device testing, vision (OpenMV)
  - Desktop applications: Tk, Qt, GTK, wxPython
  - Alternate implementations: MicroPython (embedded), Nuitka (compiled Python), Brython (JS Python), Cython (restricted data set compilation)

Thanks

- Thanks to Big Room Studios for this great space

- Thanks to Hack Portland for sponsoring
  - Please donate to Hack Portland on Open Collective, if you can
    - It helps with Pizza and creating new interest groups

Resources

- Python Hack Night Github
  - Its set up, code coming soon..

- Python Hack Night Trello
  - Not set up yet, good place to track talks, projects, etc?

- Hack Portland Web Site
  - Hack Portland Slack link (#general, #random, etc)
  - Links for donating to Hack Portland Open Collective

December Bonus

- Work through an Advent of Code
  - http://adventofcode.com