Smart Living Forerunner – Smart Plug+

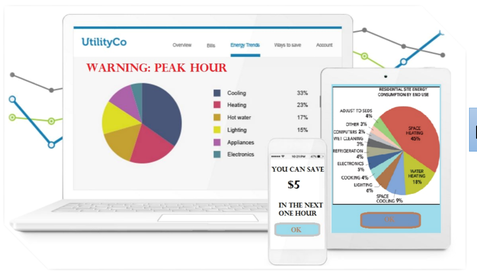
Haipeng Lu. Gan Zhang, Mengjia Kang *Instructors: Larry Henschen, Goce Trajcevski*

Introduction

The use of more home appliances means not only more convenience but also more risk caused by improper use or malfunction of appliances which may potentially result in fire later on. A complete smart-plug system, which includes real-time monitoring, remote control and abnormal warning notifications to control home appliances and save energy, is surely required.

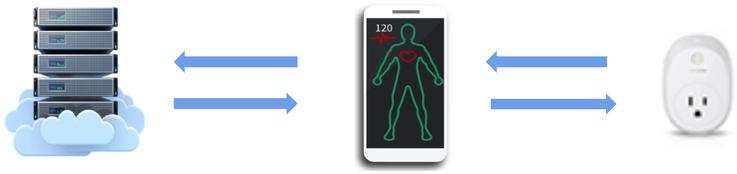
Significance

- Cloud-enabled power outlet
- Turn on and off remotely
- Offer energy monitoring
- Diagnose abnormal status
- Monthly electricity bills display

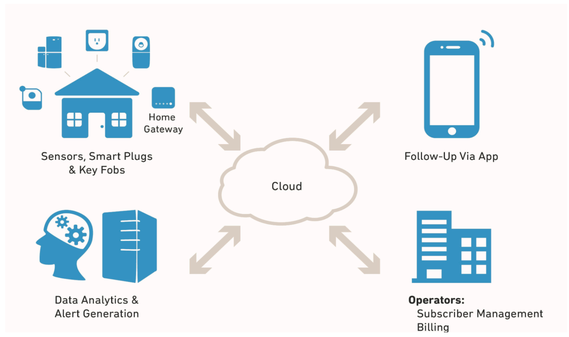


Monthly electricity bills display

Methodology



- Server-Cloud-Hold the records (Current time, current, voltage, power, device type) using MySQL
- Gateway-Send requests and display the information of the electrical appliances
- Device-Measure and collect data, receive commands from portable devices



Python Code

```python
def check_register(string):
    res = 0
    cmd = '{"system":{"get_sysinfo":{}}}'
    try:
        sock_tcp = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        sock_tcp.connect((socket_ip, port))
        sock_tcp.send(encrypt(cmd))
        data = sock_tcp.recv(2048)
        sock_tcp.close()
        res = 1
        return res
    except socket.error:
        quit("Cound not connect to host " + socket_ip + ":" + str(port))
```

Server requests data from smart outlets & Sent warning at abnormal status via Wi-Fi

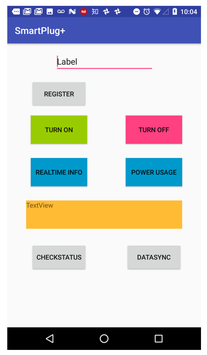
```python
def check_realtime():

    cmd = '{"emeter":{"get_realtime":{}}}'
    try:
        sock_tcp = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        sock_tcp.connect((socket_ip, port))
        sock_tcp.send(encrypt(cmd))
        data = sock_tcp.recv(2048)
        sock_tcp.close()
    except socket.error:
        quit("Cound not connect to host " + socket_ip + ":" + str(port))
    msg = decrypt(data[4:])
    rt_current = Decimal(re.findall(r"current\":(.+?),",msg)[0])
    rt_voltage = Decimal(re.findall(r"voltage\":(.+?),",msg)[0])
    rt_power = Decimal(re.findall(r"power\":(.+?),",msg)[0])
    rt_res = []
    rt_res.append(rt_current)
    rt_res.append(rt_voltage)
    rt_res.append(rt_power)
    return rt_res
```

Get Real-time Usage, Current, Voltage, Power

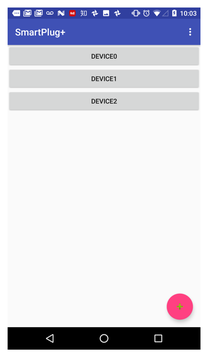
```python
def check_status_name():
    #-----------------
    # Get realtime power
    cmd = '{"emeter":{"get_realtime":{}}}'
    try:
        sock_tcp = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        sock_tcp.connect((socket_ip, port))
        sock_tcp.send(encrypt(cmd))
        data = sock_tcp.recv(2048)
        sock_tcp.close()
    except socket.error:
        quit("Cound not connect to host " + socket_ip + ":" + str(port))
    msg = decrypt(data[4:])
    realtime_power = Decimal(re.findall(r"power\":(.+?),",msg)[0])
```

Check status of the Appliance

Application User Interface



Options to Turn on & off, Check Real-time info &Power Usage



Labeled Appliances

Future Works

- Identify different devices
- Solve compatibility issues among smartphones with different OS platforms.

Powered by GitHub android MySQL tp-link

Contact info : <haipenglu2017@u.northwestern.edu> <ganzhang2016@u.northwestern.edu> <mengjiakang2017@u.northwestern.edu>