

# به نام خدا

گزارش فاز سوم پروژه  
سیستم های نهفته و بی درنگ  
موضوع پروژه: دزدگیر ماشین

پریسا علائی، 97522175  
غزل زمانی نژاد، 97522166

در فاز پایانی پروژه به جمع آوری داده پرداختیم. در این پروژه 3 دسته اطلاعات به سرور ارسال می شود که به بررسی آنها می پردازیم:

- وضعیت ماشین: topic آن 97522175/car\_state است. زمانی که از فندکی ماشین ولتاژ دریافت شود، Car is ON به همراه زمان (که از سرور NTP آن را دریافت کرده ایم) به سرور ارسال می شود به طور مثال:

5444	Thu Jun 30 07:27:34 2022	97522175/car_state	Thursday, 30 June 2022, 11:57:34 Car is ON
------	--------------------------	--------------------	--

و در صورتی که ماشین خاموش باشد:

5438	Thu Jun 30 07:27:31 2022	97522175/car_state	Thursday, 30 June 2022, 11:57:29 Car is OFF
------	--------------------------	--------------------	---

- وضعیت دزدگیر: topic آن 97522175/alarm\_state است. زمانی که ماشین خاموش باشد، دزدگیر روشن است و Alarm is ON به همراه زمان (که از سرور NTP آن را دریافت کرده ایم) به سرور ارسال می شود به طور مثال:

5448	Thu Jun 30 07:27:37 2022	97522175/alarm_state	Thursday, 30 June 2022, 11:57:36 Alarm is ON
------	--------------------------	----------------------	--

و در صورتی که دزدگیر غیرفعال باشد:

5451	Thu Jun 30 07:27:39 2022	97522175/alarm_state	Thursday, 30 June 2022, 11:57:39 Alarm is OFF
------	--------------------------	----------------------	---

- وضعیت بازر: topic آن 97522175/buzzer\_state است. سنسور به صورت مداوم مشغول به اندازه گیری محیط اطراف است. زمانی که مقدار شتاب اندازه گیری شده از مقدار threshold بیشتر شود، زنگ اخبار بوق می زند و در نتیجه Buzzer is beeping! به سرور ارسال می شود به طور مثال:

6899	Thu Jun 30 11:09:05 2022	97522175/buzzer_state	Buzzer is beeping!
------	--------------------------	-----------------------	--------------------

و در صورتی که بازر غیرفعال باشد:

6890	Thu Jun 30 11:08:36 2022	97522175/buzzer_state	Buzzer is OFF
------	--------------------------	-----------------------	---------------

همچنین کاربر می تواند با ارسال یک مقدار به سرور، رله را خاموش / روشن کند. topic آن 97522175/ONOFF\_FEED است. برای روشن کردن سوییچ، relay=1 و برای خاموش کردن relay=0 به سرور ارسال می کنیم. به طور مثال:

12046	Sat Jul 2 13:25:20 2022	97522175/ONOFF_FEED	relay=0
12237	Sat Jul 2 13:47:09 2022	97522175/ONOFF_FEED	relay=1

پس از کار کردن دزدگیر به مدت حدودا یک هفته، فایل CSV را دریافت کرده و اطلاعات آن را مورد بررسی قرار می دهیم. برای دانلود آن از <http://45.149.77.235:8000/download> استفاده کردیم. ابتدا با استفاده از csv\_reader فایل را می خوانیم:

```
1 import csv
2 import matplotlib.pyplot as plt
3
4
5 with open('97522175.csv') as f:
6     csvreader = csv.reader(f)
7     header = []
8     header = next(csvreader)
9     print(header)
10
11     rows = []
12     for row in csvreader:
13         rows.append(row)
14
15 print(rows[0])
```

```
['text/csvnumber', 'time', 'topic', 'data']
['492', 'Tue Jun 28 11:59:17 2022', '97522175/car_state', 'Car is OFF']
```

سپس یک دیکشنری تشکیل می دهیم، تعداد سطرهای مشابه را می شماریم و مقادیر را در دیکشنری ذخیره می کنیم.

```

1 topics = ['97522175/car_state', '97522175/alarm_state', '97522175/buzzer_state']
2 data = {}
3 for t in topics:
4     data[t] = {}
5     data[t]['on'] = 0
6     data[t]['off'] = 0

```

```

1 for row in rows:
2     topic = row[2]
3     if topic in topics:
4         if 'OFF' in row[3]:
5             data[topic]['off'] += 1
6         else:
7             data[topic]['on'] += 1

```

خروجی به صورت زیر است:

```

{'97522175/car_state': {'on': 11023, 'off': 13669},
'97522175/alarm_state': {'on': 10376, 'off': 10922},
'97522175/buzzer_state': {'on': 1348, 'off': 19720}}

```

	on	off
Car state	11023	13669
Alarm state	10376	10922
Buzzer state	1348	19720

سپس مجموع دقایقی که ماشین روشن بوده را محاسبه می کنیم. برای این کار یک مقدار Boolean تعریف می کنیم که در ابتدا false است. اگر به سطر Car is ON برسیم و مقدار بولین غلط باشد یعنی ماشین به تازگی روشن شده، پس زمان آن را در یک متغیر ذخیره می کنیم. اگر به سطر Car is OFF برسیم و مقدار بولین صحیح باشد یعنی ماشین به تازگی خاموش شده. زمان آن را از زمان قبلی کم می کنیم و به مقدار مجموع اضافه می کنیم و در نهایت آن را چاپ می کنیم.

```

1 import datetime
2
3 # calculate the total minutes that car was on
4 state_on = False
5 total_min = 0
6
7 for row in rows:
8     topic = row[2]
9     if topic == topics[0]:
10         if (not state_on) and ('ON' in row[3]): # the car has just started working
11             date = row[1].split()
12             time = date[3].split(':')
13             time1 = datetime.datetime(2022, 7, int(date[2]), int(time[0]), int(time[1]), int(time[2]))
14             # print(time1)
15             state_on = True
16
17         if state_on and ('OFF' in row[3]): # the car has just stopped working
18             date = row[1].split()
19             time = date[3].split(':')
20             time2 = datetime.datetime(2022, 7, int(date[2]), int(time[0]), int(time[1]), int(time[2]))
21             total_min += (time2 - time1).total_seconds() / 60
22             state_on = False

```

خروجی:

The car was on for 6408.35 minutes totally.

تعداد دفعاتی که کاربر سویچ را خاموش/روشن کرده را نیز می توانیم بشماریم. برای این کار بر روی داده ها iterate می کنیم و هر سطری که topic آن 97522175/ONOFF\_FEED باشد بررسی می کنیم.

```

1 # count commands sent to server
2 data['97522175/ONOFF_FEED'] = {}
3 data['97522175/ONOFF_FEED']['on'] = 0
4 data['97522175/ONOFF_FEED']['off'] = 0
5
6 for row in rows:
7     topic = row[2]
8     if topic == '97522175/ONOFF_FEED':
9         if row[3] == 'relay=0':
10             data['97522175/ONOFF_FEED']['off'] += 1
11         if row[3] == 'relay=1':
12             data['97522175/ONOFF_FEED']['on'] += 1

```

خروجی:

User turned the switch off 24 times  
and turned it on 30 times.

### 3 دسته اطلاعات را در قالب pie chart 3 رسم می کنیم.

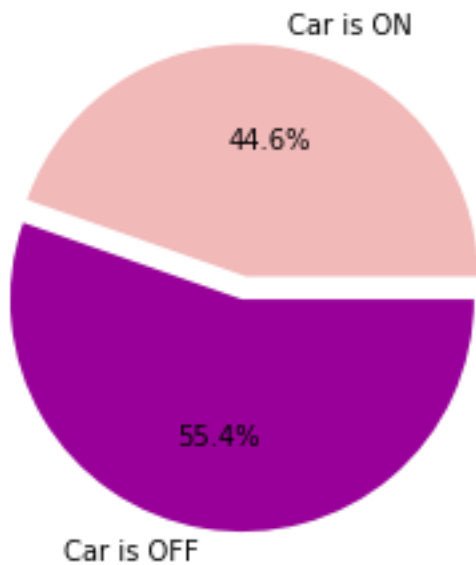


Figure 1 - car state pie chart

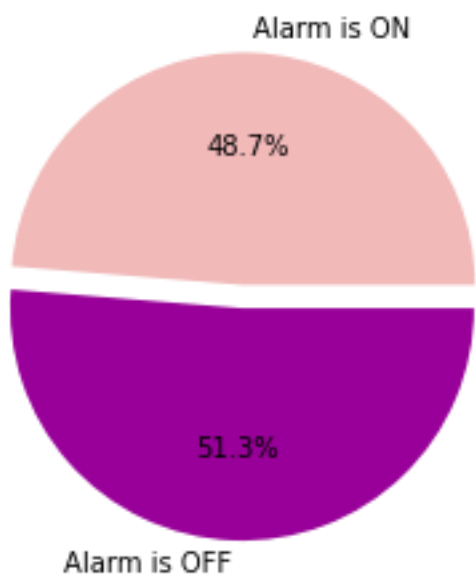
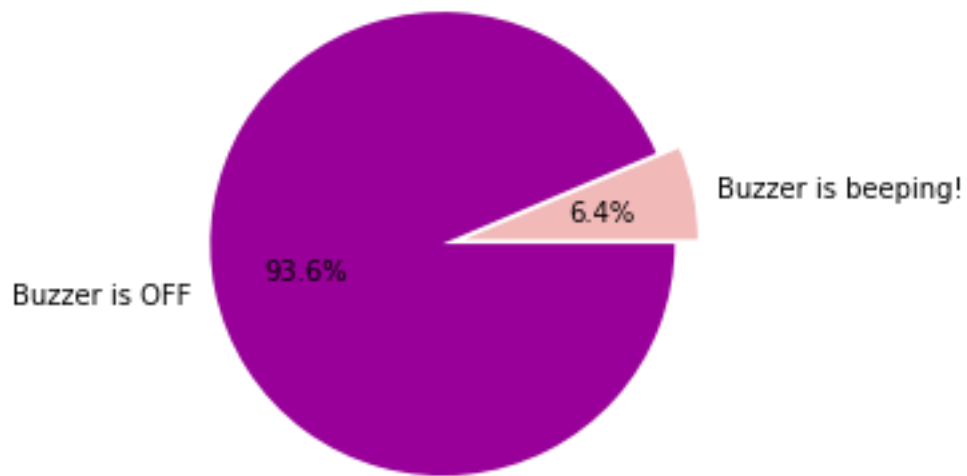


Figure 2 - alarm state pie chart



*Figure 3 - buzzer state pie chart*