

#Heisenbug

ТЕСТИРОВАНИЕ ИНТЕРНЕТА ЛИФТОВ

Елена Колпакова





AKQA

@HeleneKolpakova

elena.kolpakova@akqa.com

АНАТОМИЯ IOT ЛИФТА



ПРИЛОЖЕНИЯ

Lift analytics

Time range: 14 days / by day

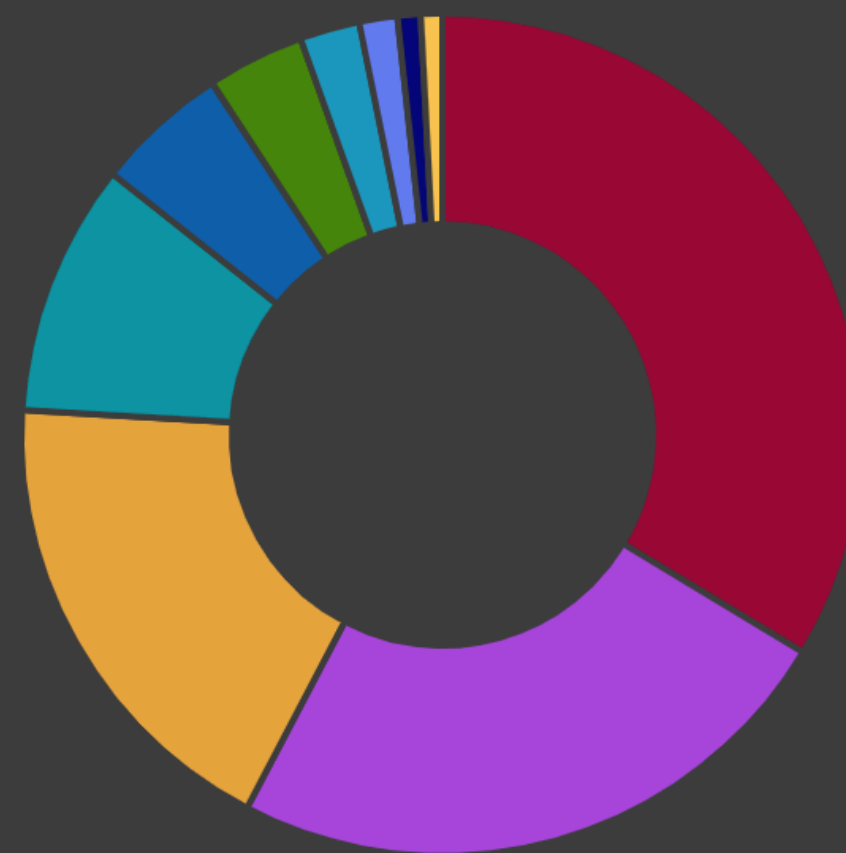
Runs

22,901

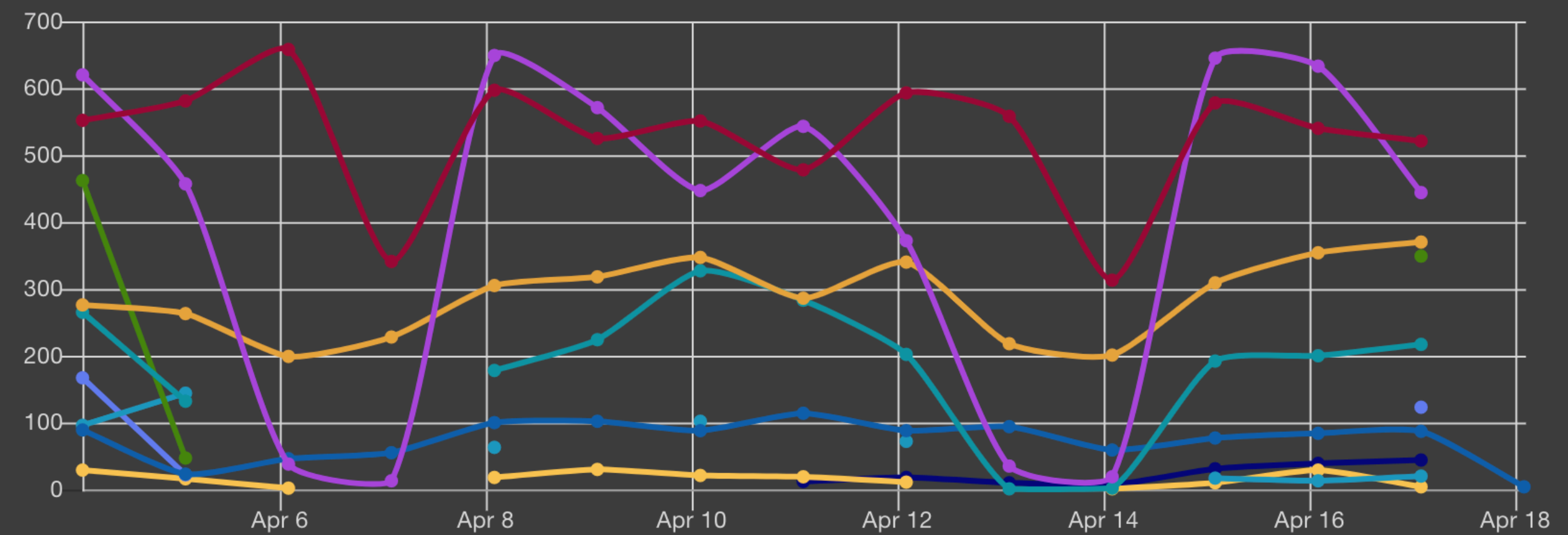
Door cycles

26,005

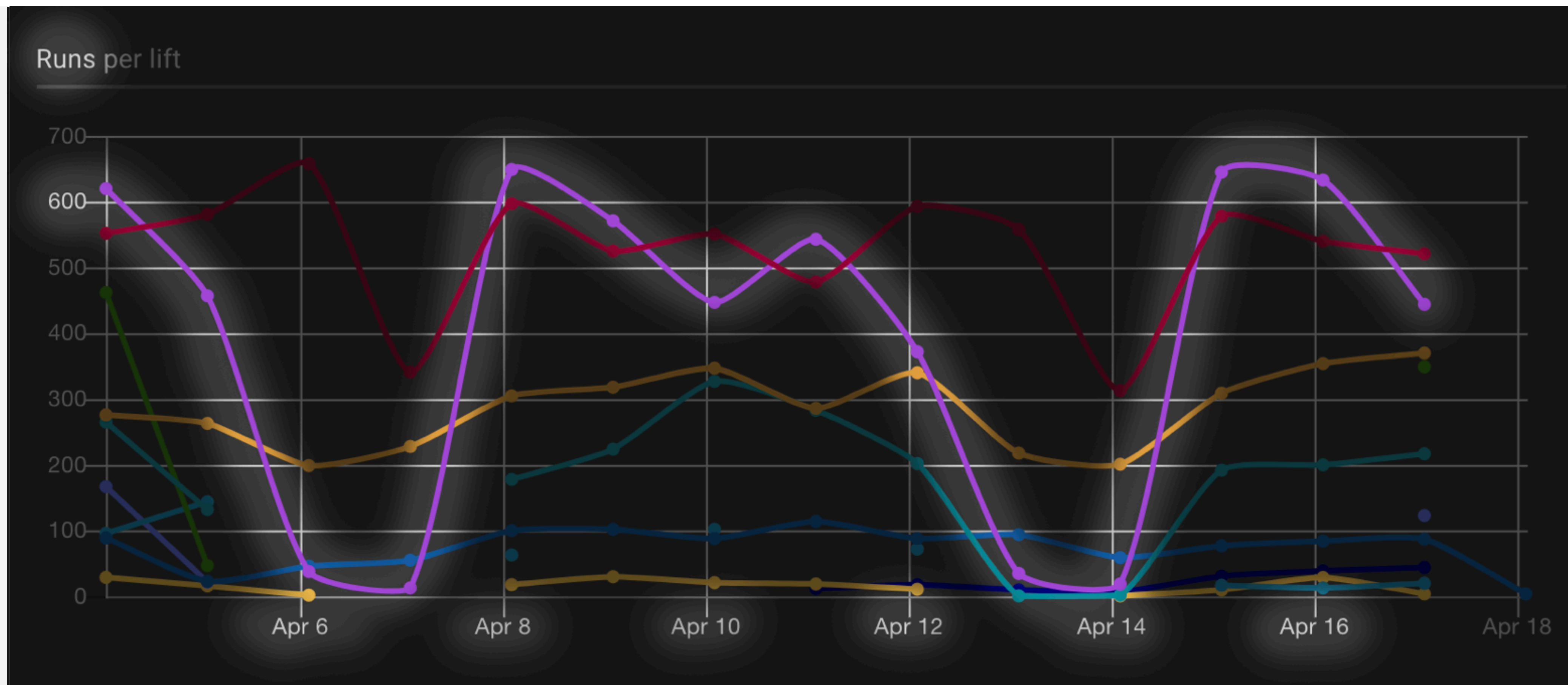
Runs per lift



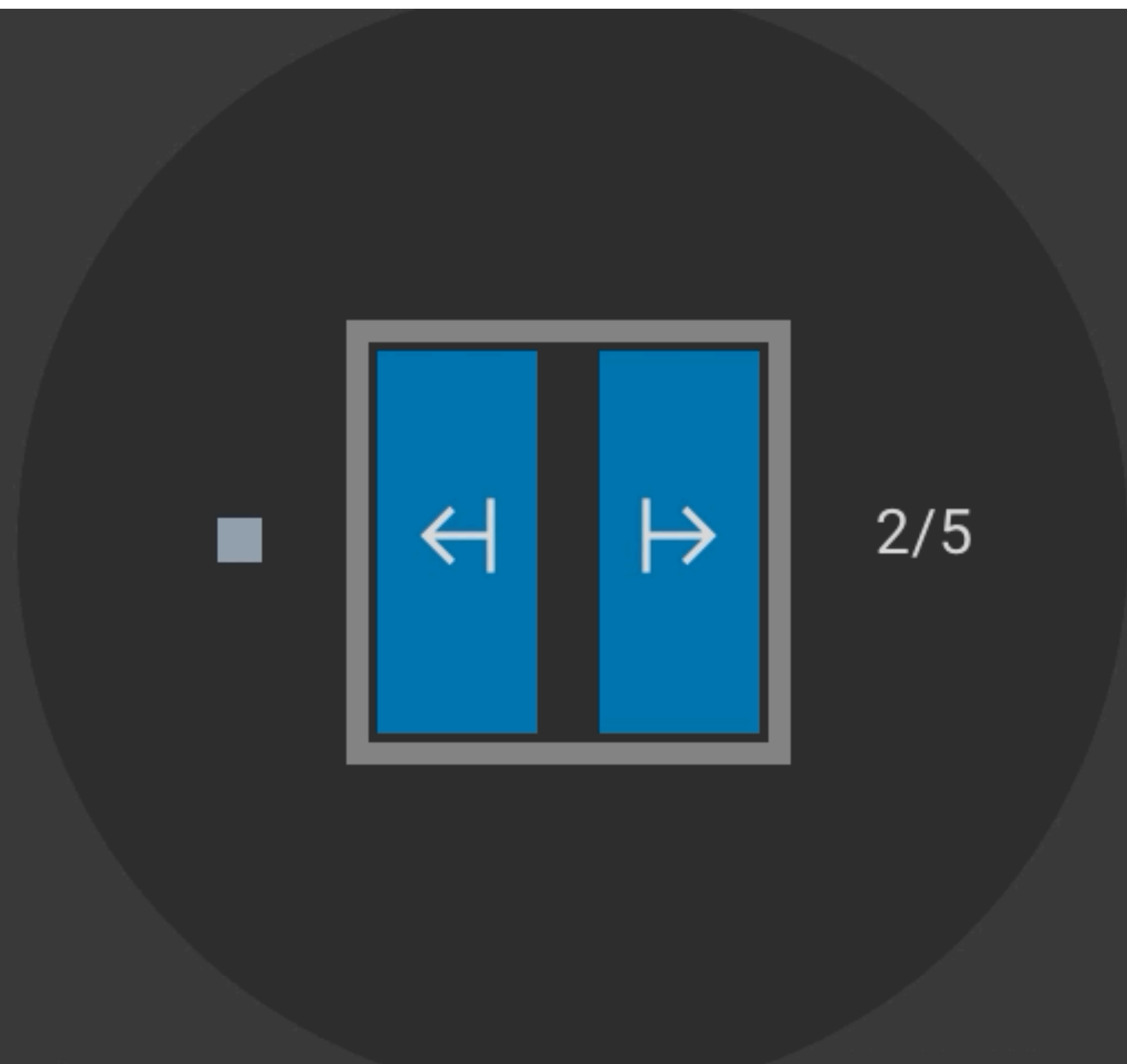
Runs per lift



ПРИЛОЖЕНИЯ



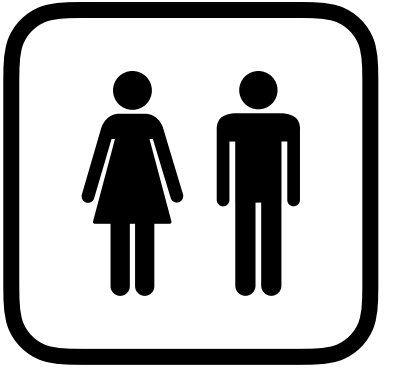
ПРИЛОЖЕНИЯ



ЛИФТЫ



ЛИФТЫ



Aufzug Nr. 76 Tragkraft 8000 kg

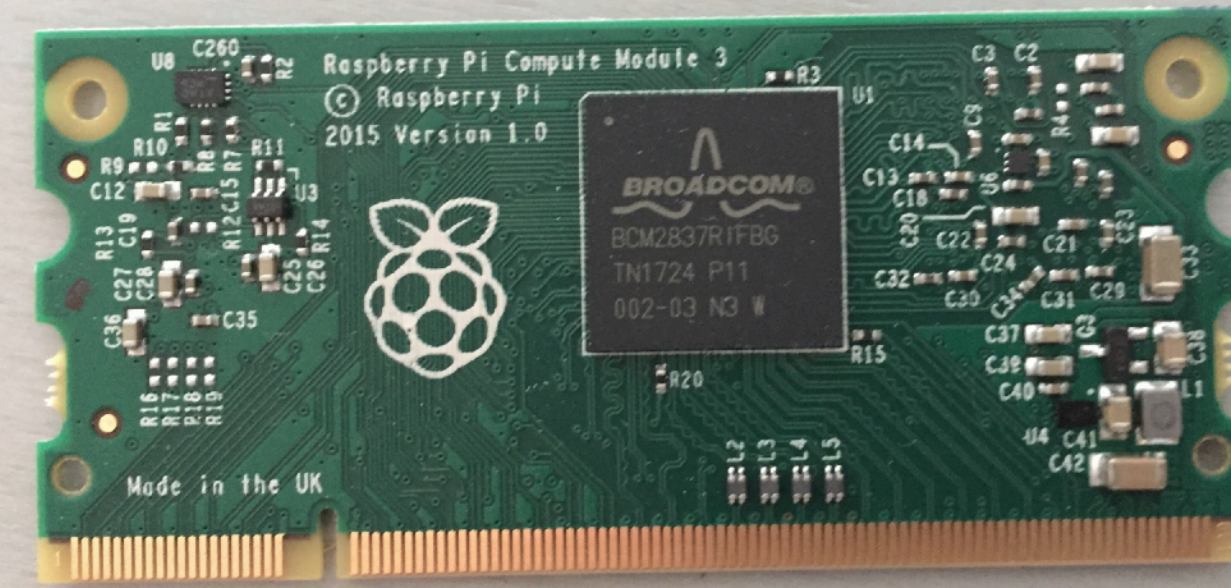
ЛИФТ - КОНЕЧНЫЙ АВТОМАТ



EDGE GATEWAY



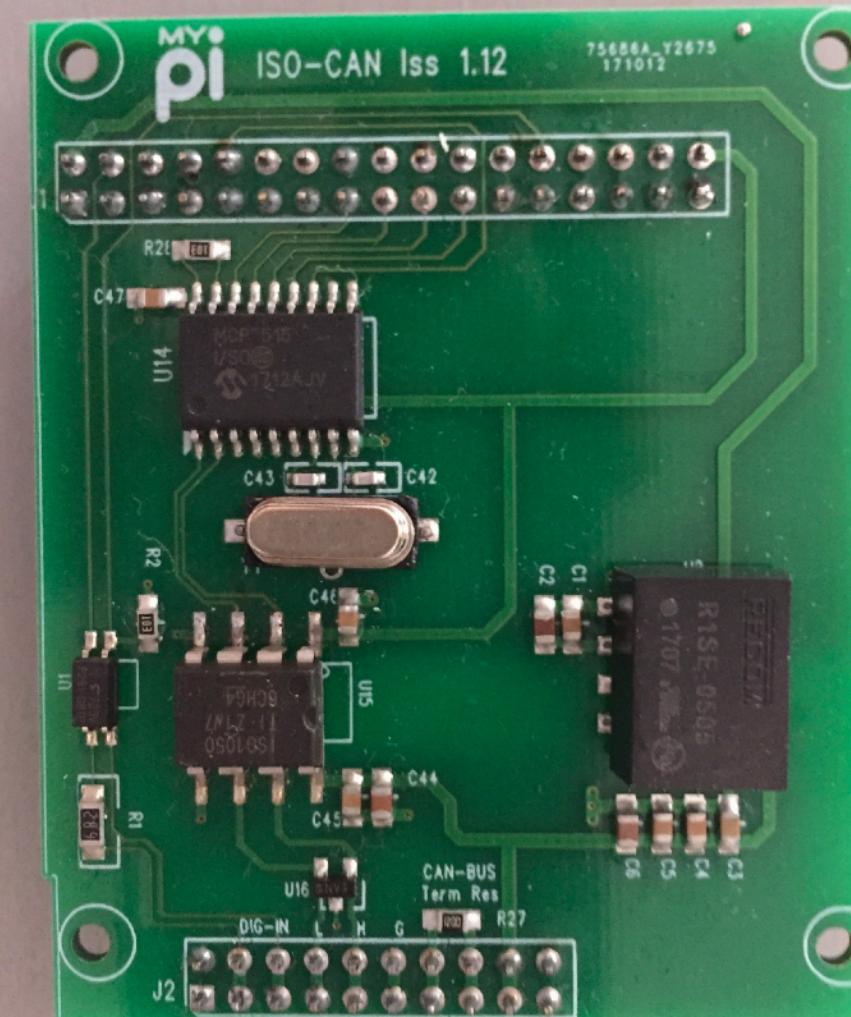
- **Raspberry Pi**
- **CAN-модуль**
- **3G/LTE-модем**
- **SIM-карта**
- **.js приложения**



EDGE GATEWAY



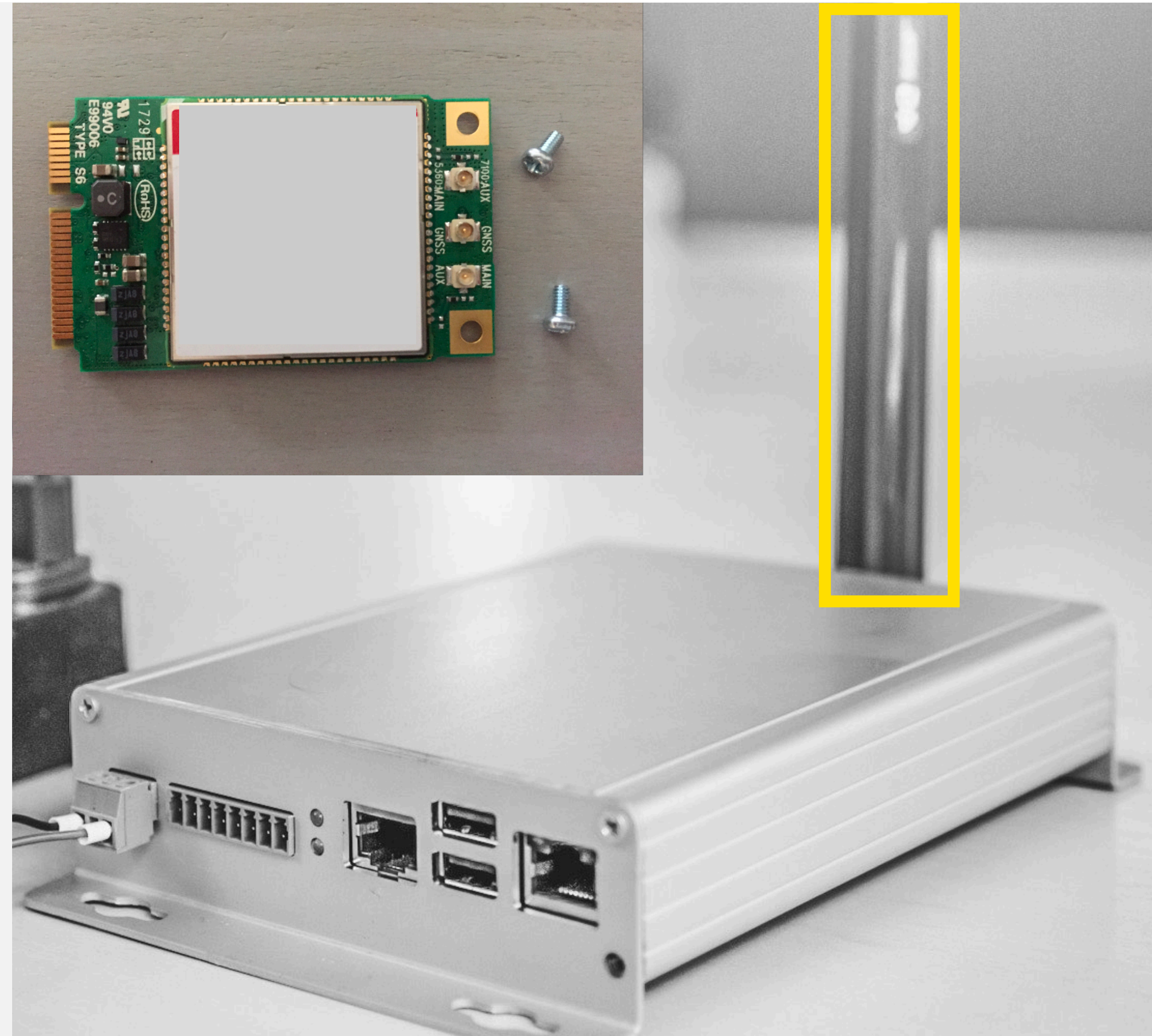
- Raspberry Pi
- CAN-модуль
- 3G/LTE-модем
- SIM-карта
- .js приложения



EDGE GATEWAY



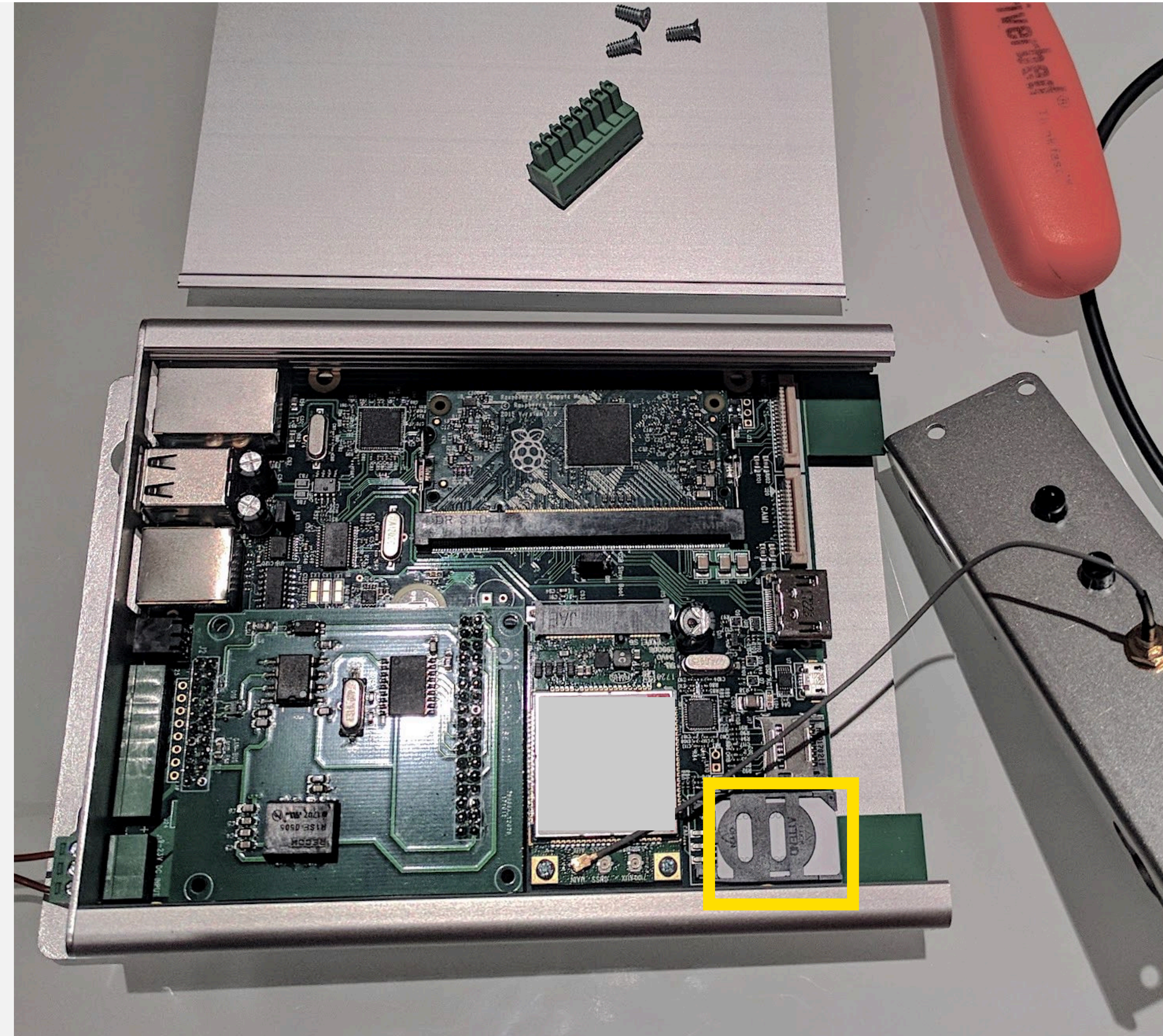
- Raspberry Pi
- CAN-модуль
- 3G/LTE-модем
- SIM-карта
- .js приложения



EDGE GATEWAY



- Raspberry Pi
- CAN-модуль
- 3G/LTE-модем
- SIM-карта
- .js приложения



EDGE GATEWAY



- Raspberry Pi
- CAN-модуль
- 3G/LTE-модем
- SIM-карта
- **.js приложения**

JS



ОБЛАКО



- **AWS IoT**
- **Lambda**
- **Elastic Search**



ОБЛАКО



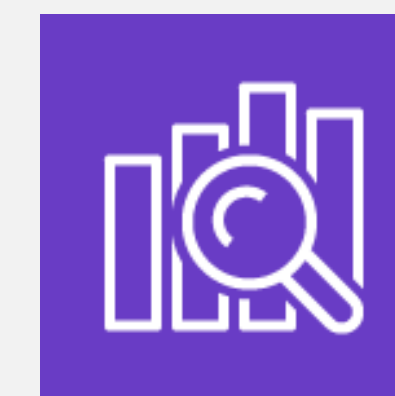
EDGE
GATEWAY



AWS IoT



LAMBDA



ELASTIC
SEARCH



IoT ПЛАТФОРМА

ИОТ ПЛАТФОРМА

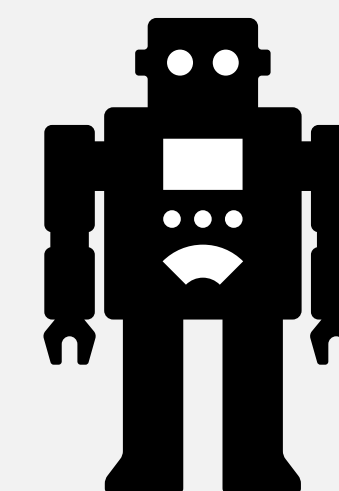


ЛИФТ

УДАЛЁННОЕ УПРАВЛЕНИЕ

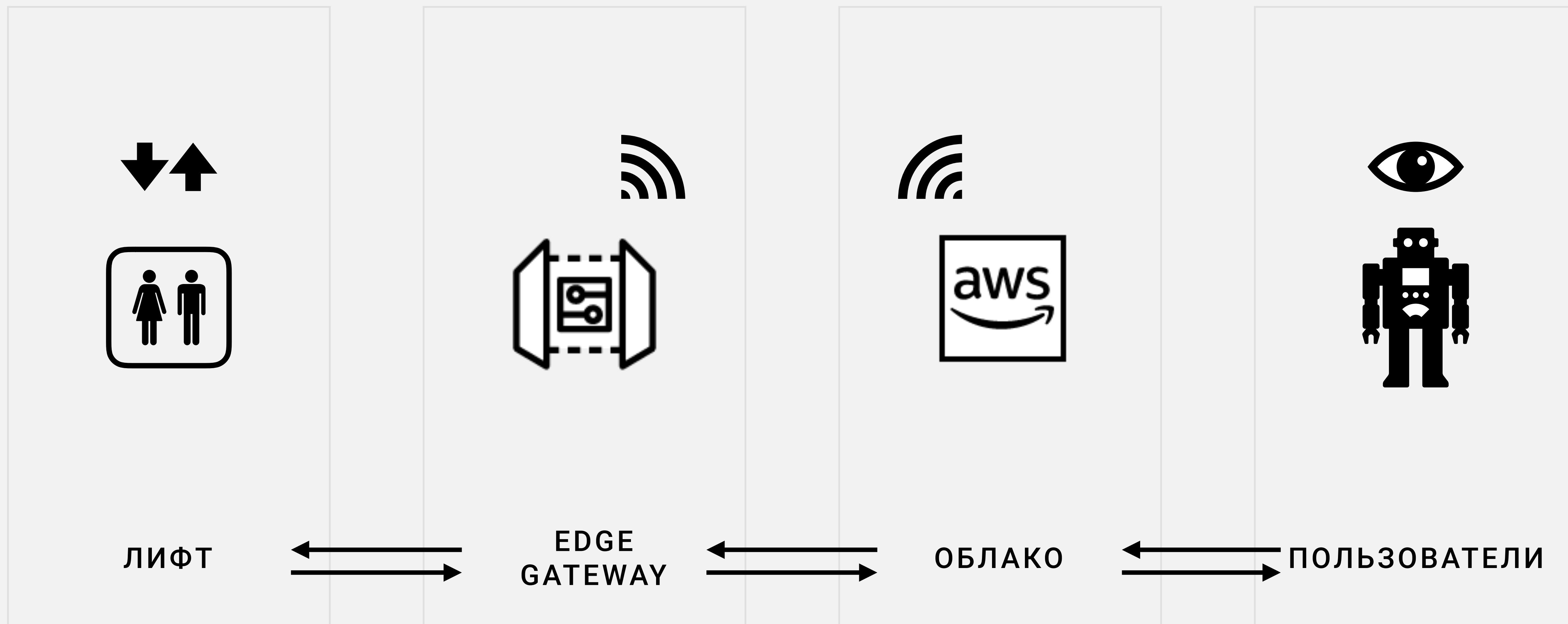


СБОР ДАННЫХ



ПОЛЬЗОВАТЕЛИ

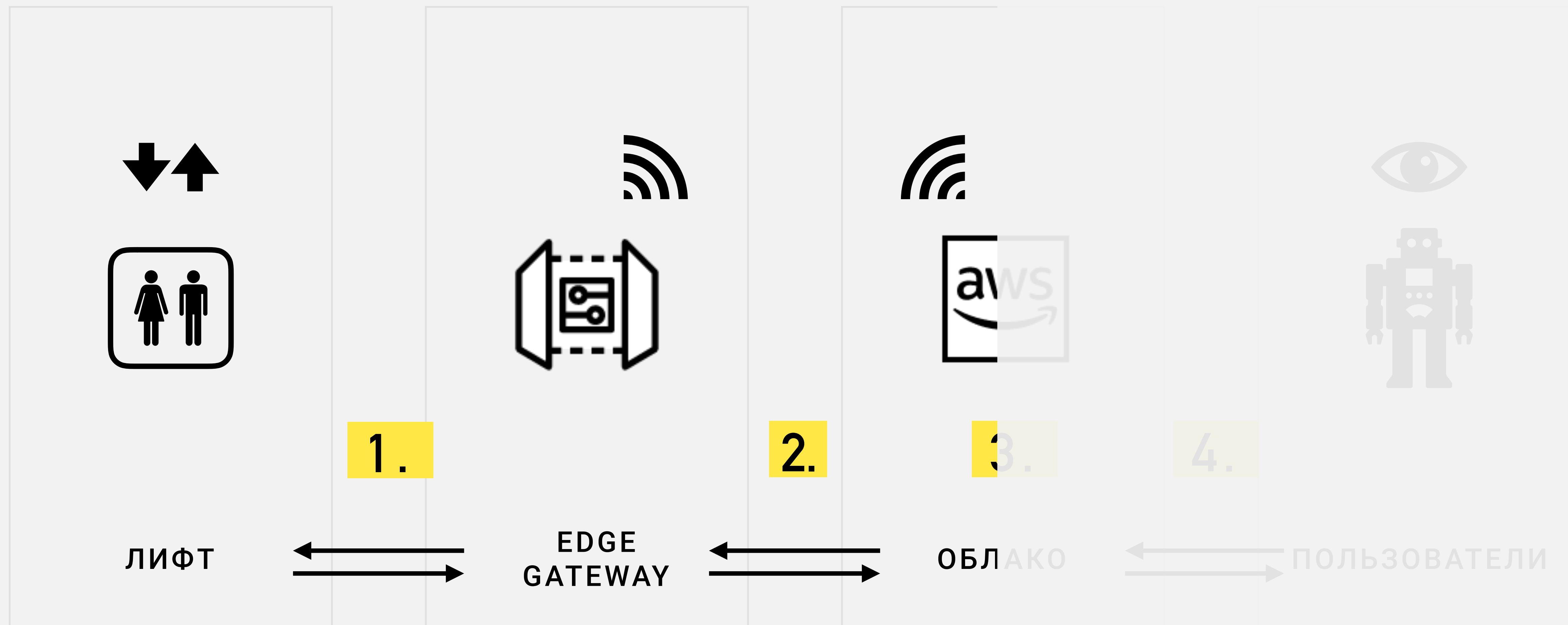
ІОТ ПЛАТФОРМА



ФУНКЦИИ ИОТ



ІОТ ПЛАТФОРМА



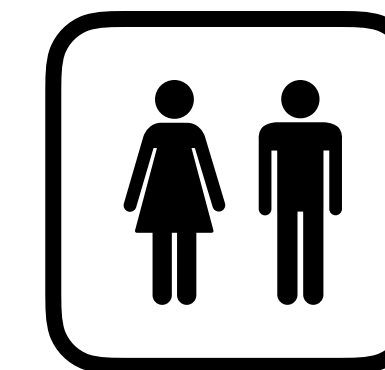
ТЕСТИРОВАНИЕ КАК ЭТО БЫЛО



ТЕСТ-ЛИФТ



ТЕСТ-ЛИФТ

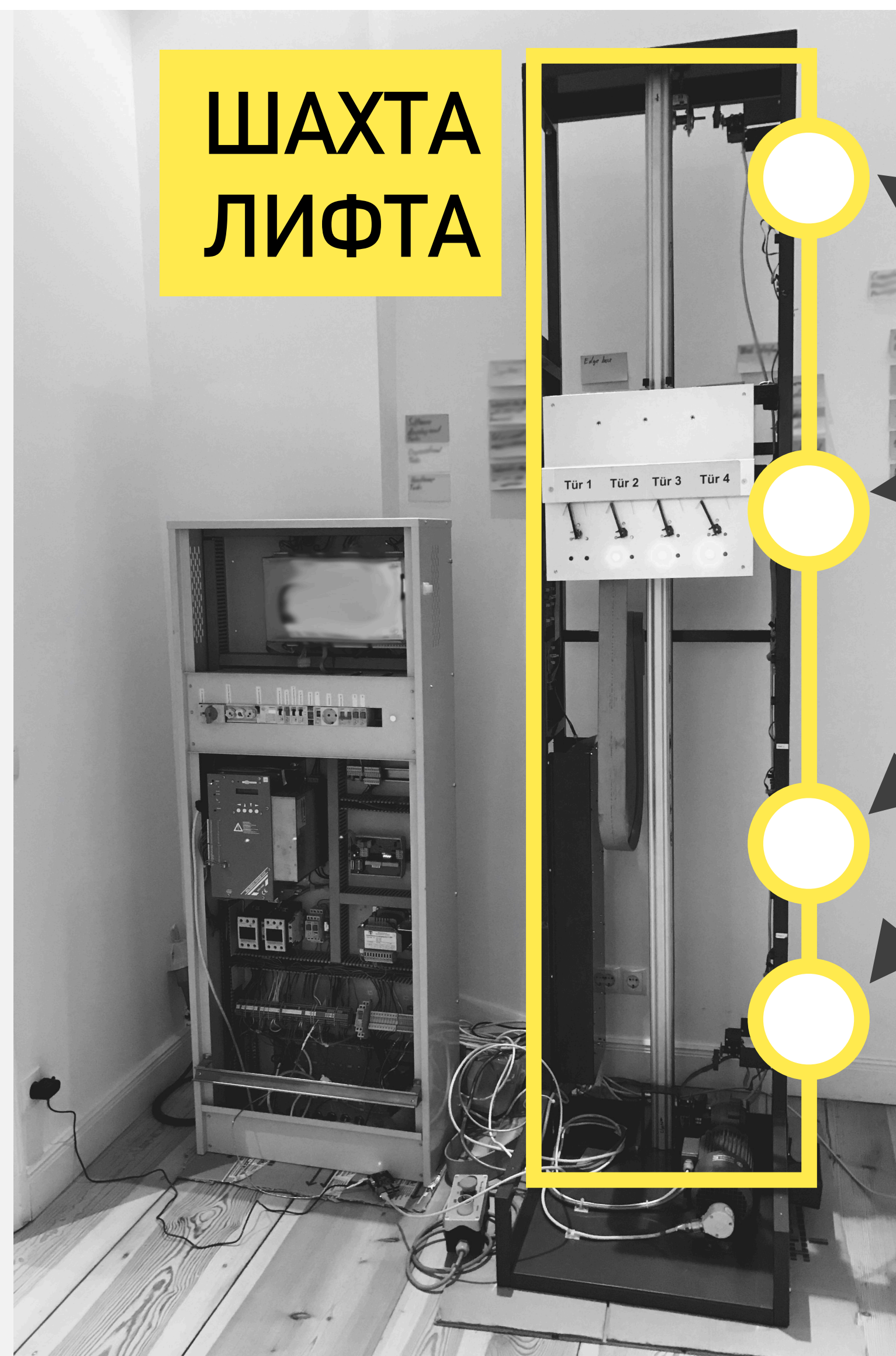
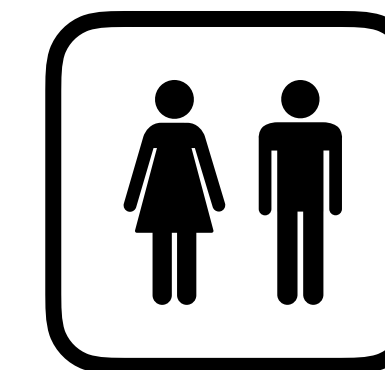


КОНТРОЛЛЕР

Tür 1 Tür 2 Tür 3 Tür 4

**КАБИНА
ЛИФТА**

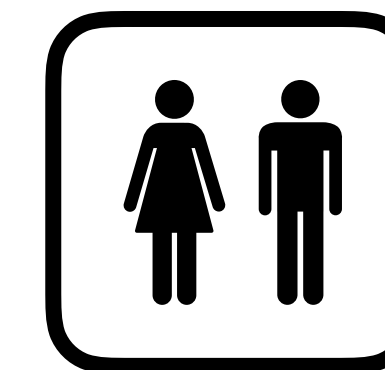
ТЕСТ-ЛИФТ



ШАХТА
ЛИФТА

ДАТЧИКИ
ЭТАЖЕЙ

ТЕСТ-ЛИФТ



**EDGE
GATEWAY**

**ДВЕРИ
ЛИФТА**

ДВИГАТЕЛЬ



(2019-04-15 11:40:01.078778)	vcan0	42C	[8]	7F 4C 00 02 00 00 7F 4C
(2019-04-15 11:40:01.098356)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.128564)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09
(2019-04-15 11:40:01.174977)	vcan0	720	[8]	02 02 1F 00 22 01 00 00
(2019-04-15 11:40:01.178449)	vcan0	42C	[8]	7F 40 00 02 00 00 7F 40
(2019-04-15 11:40:01.198023)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.228214)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09
(2019-04-15 11:40:01.274491)	vcan0	720	[8]	02 02 1F 00 22 01 00 00
(2019-04-15 11:40:01.277946)	vcan0	42C	[8]	7F 35 00 02 00 00 7F 35
(2019-04-15 11:40:01.297630)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.327884)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09
(2019-04-15 11:40:01.375226)	vcan0	720	[8]	02 02 1F 00 22 01 00 00
(2019-04-15 11:40:01.378684)	vcan0	42C	[8]	7F 29 00 02 00 00 7F 29
(2019-04-15 11:40:01.398290)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.428484)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09
(2019-04-15 11:40:01.474712)	vcan0	720	[8]	02 02 1F 00 22 01 00 00
(2019-04-15 11:40:01.478214)	vcan0	42C	[8]	7F 1D 00 02 00 00 7F 1D
(2019-04-15 11:40:01.498907)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.528112)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09
(2019-04-15 11:40:01.574527)	vcan0	720	[8]	02 02 1F 00 22 01 00 00
(2019-04-15 11:40:01.578087)	vcan0	42C	[8]	7F 11 00 02 00 00 7F 11
(2019-04-15 11:40:01.598063)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.627347)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09
(2019-04-15 11:40:01.674800)	vcan0	720	[8]	02 02 1F 00 22 01 00 00
(2019-04-15 11:40:01.678183)	vcan0	42C	[8]	7F 05 00 02 00 00 7F 05
(2019-04-15 11:40:01.698617)	vcan0	42B	[8]	0A 33 40 30 03 FF 00 00
(2019-04-15 11:40:01.727541)	vcan0	446	[8]	03 01 02 00 0F 00 A3 09

ВРЕМЯ

КОМАНДА

ЗНАЧЕНИЕ
(до 8 байт)

```
:40:03.872407) vcan0 720 [8] 02 02 1F 00 22 01 00 00
:40:03.875347) vcan0 42C [8] 7E 01 00 01 00 00 7E 01
:40:03.900713) vcan0 42B [8] 0A 33 40 30 03 FF 00 00
:40:03.925121) vcan0 446 [8] 03 01 02 00 0F 00 A3 09
:40:03.943988) vcan0 422 [0]
:40:03.945553) vcan0 7E4 [8] 80 0A 80 0B 00 00 00 00
:40:03.946006) vcan0 441 [6] 00 08 0 02
:40:03.946 vcan0 6A0 [8] 00 01 0 00 00 00
:40:03.972695) vcan0 720 [8] 02 02 1 01 00 00
:40:03.975283) vcan0 42C [8] 7D F5 00 01 00 00 7D F5
:40:04.000429) vcan0 42B [8] 0B 33 40 30 03 FF 00 00
:40:04.024644) vcan0 446 [8] 04 01 02 00 0F 00 A3 09
:40:04.072347) vcan0 720 [8] 02 02 1F 00 22 01 00 00
:40:04.075005) vcan0 42C [8] 7D E9 00 01 00 00 7D E9
:40:04.101252) vcan0 42B [8] 0B 33 40 30 03 FF 00 00
:40:04.101891) vcan0 4A0 [8] 01 01 01 01 00 02 03 13
:40:04.124953) vcan0 446 [8] 04 01 02 00 0F 00 A3 09
:40:04.140443) vcan0 488 [1] 01
:40:04.172042) vcan0 720 [8] 02 02 1F 00 22 01 00 00
```

КОМАНДА

446

446

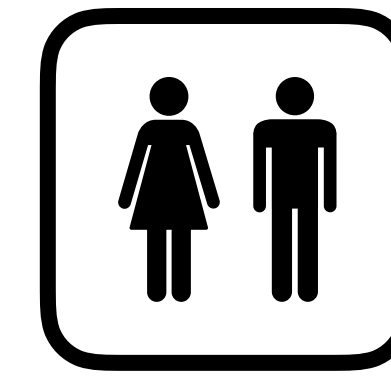
446

03

04

**СМЕНА
ПЕРВОГО
БАЙТА**

СПЕЦИФИКАЦИЯ



446 – elevator state

cansend example: 446#0000000000F00C305

B1 – ?? state: {00, 01, 02, 03, 04, 05, 06, 09}

00 – idle

01 – prepare to go

02 – moving (engine brakes off)

03 – moving outside the floor stop mark (engine full speed)

04 – braking (engine slow speed)

05 – braking stop started (engine brakes on)

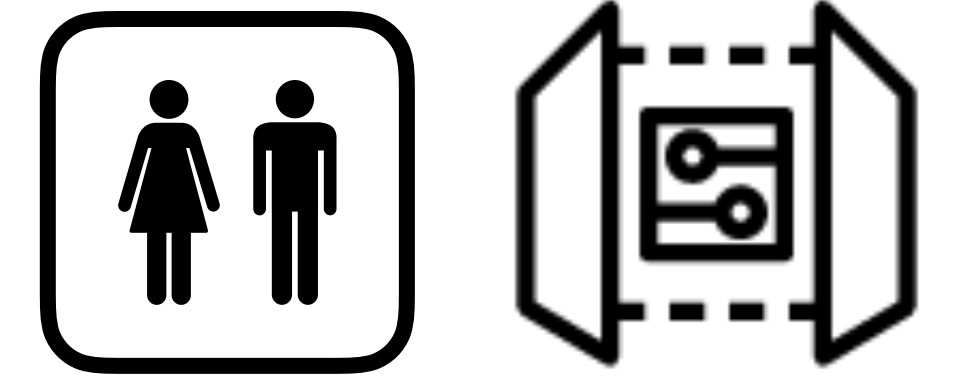
06 – braking stop stopped

09 – stopped

B2-B3 – direction: {0000, 0101, 0102}

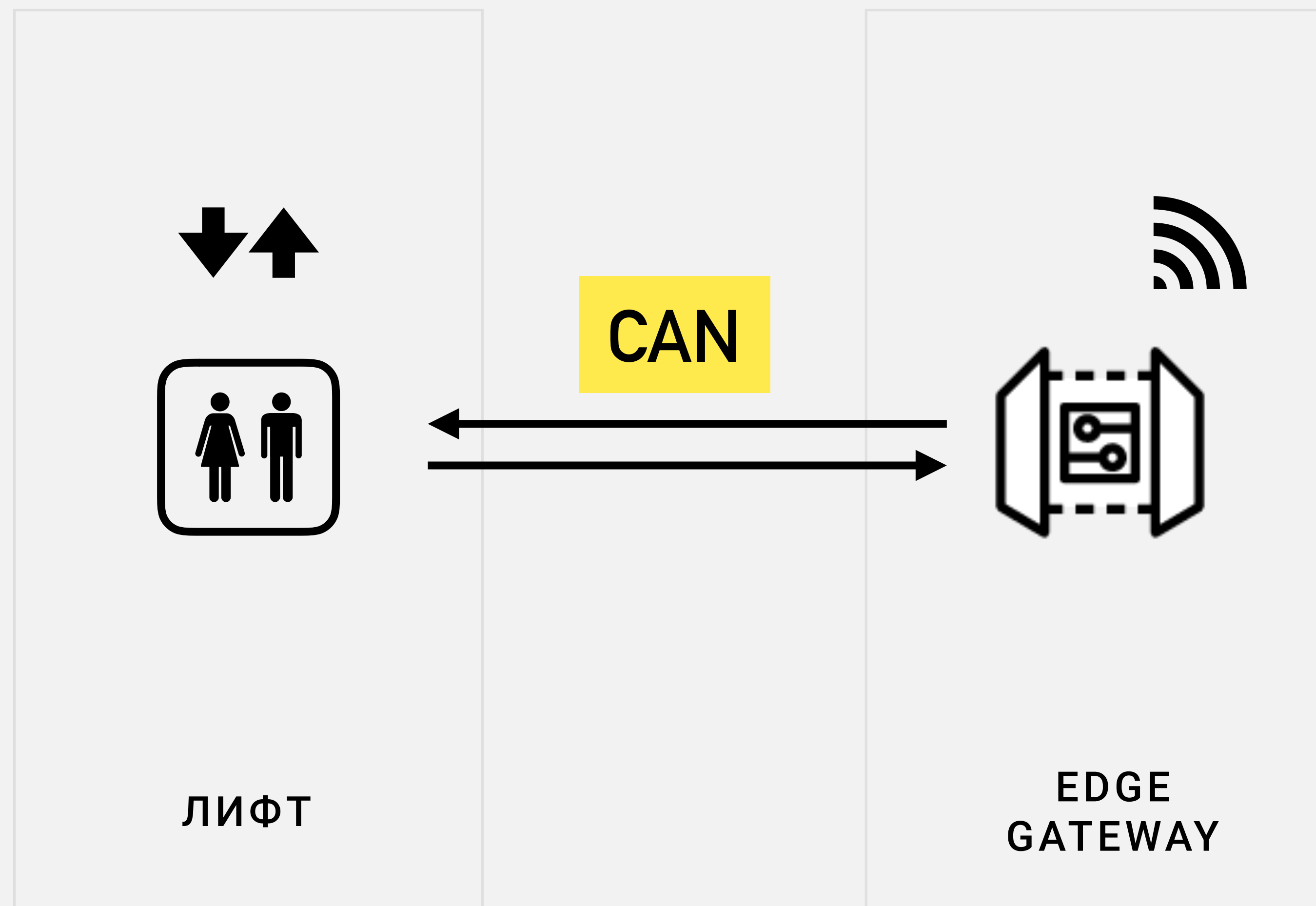
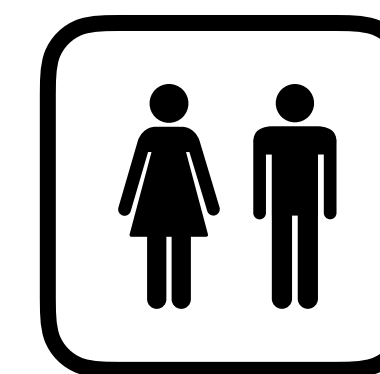


ПОБОЧНЫЙ ЭФФЕКТ



1. Изучите состояния и поведение лифта
2. Получите набор тестовых данных,
покрывающий все состояния
3. Симулятор лифта

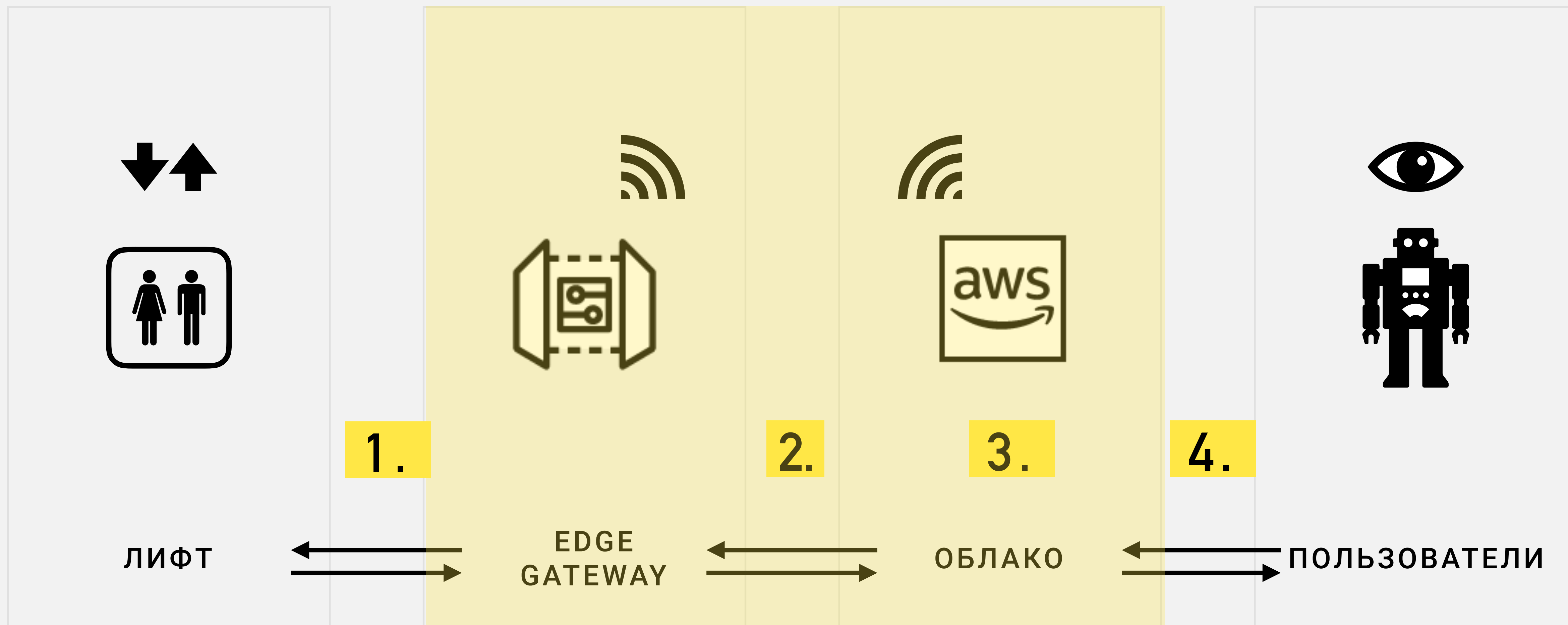
КОНВЕРТАЦИЯ ДАННЫХ



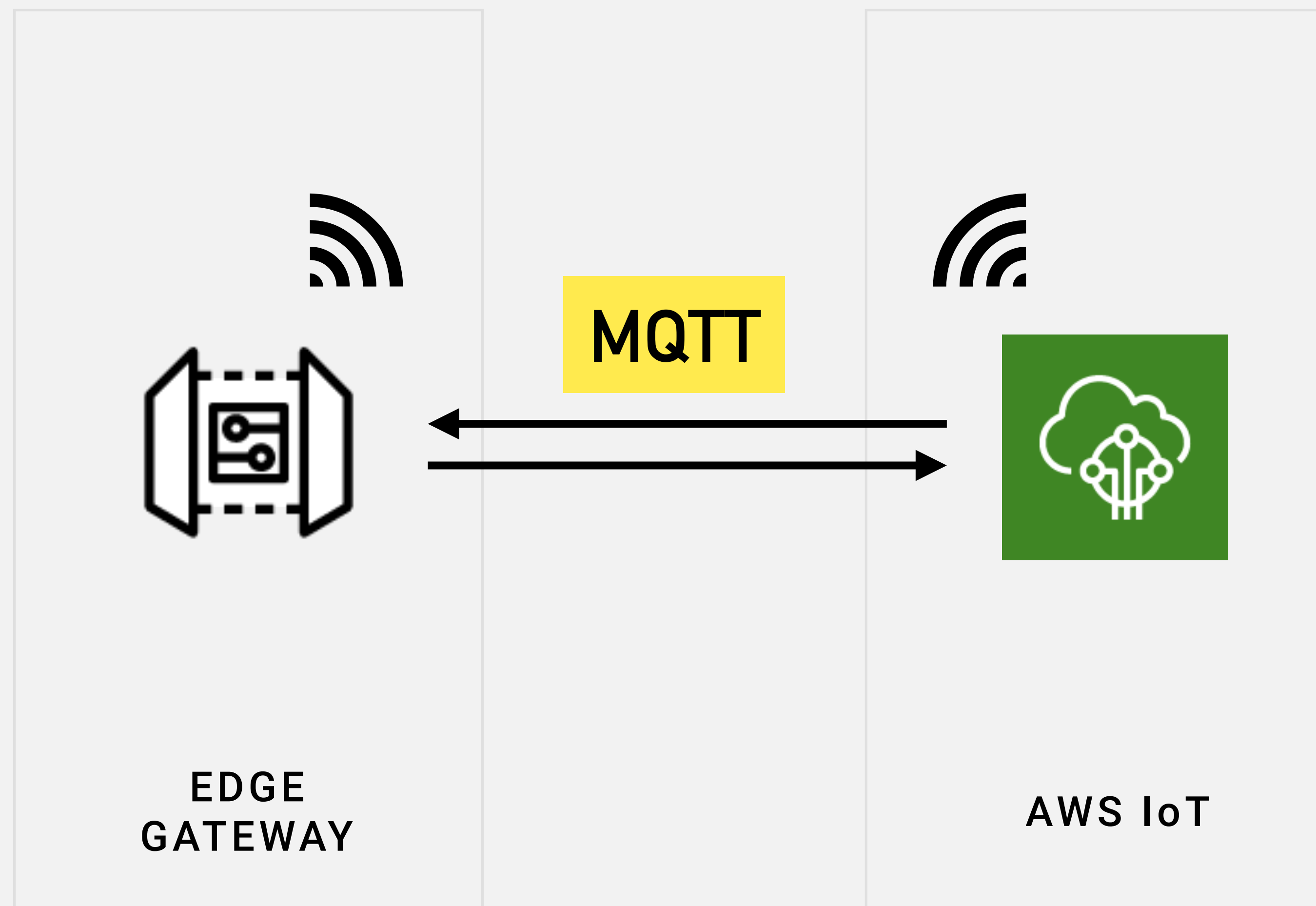
ФУНКЦИИ IOT



ФУНКЦИИ ИОТ



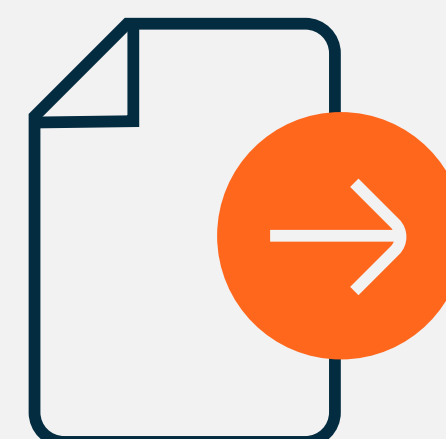
ПЕРЕСЫЛКА ДАННЫХ



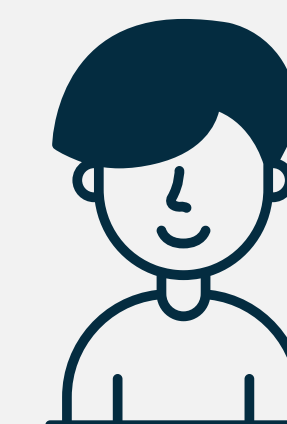
REST: МОДЕЛЬ “ЗАПРОС-ОТВЕТ”



АЛИСА



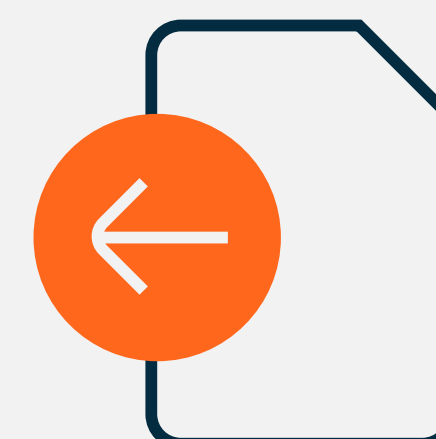
ЗАПРОС



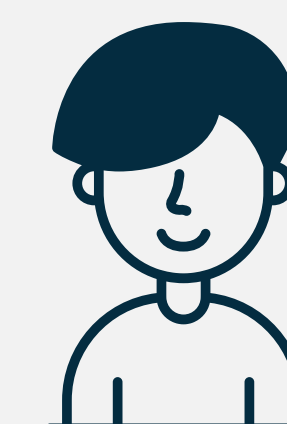
БОБ



АЛИСА



ОТВЕТ



БОБ

REST API ТЕСТЫ



1. Отправить запрос:

```
GET /endpoint/name?parameter=test
```

2. Проверить ответ:

```
1. Status: 200
```

```
2. Headers: Content-Type application/json
```

```
3. Payload: { "statusCode": 200 }
```

REST API ТЕСТЫ

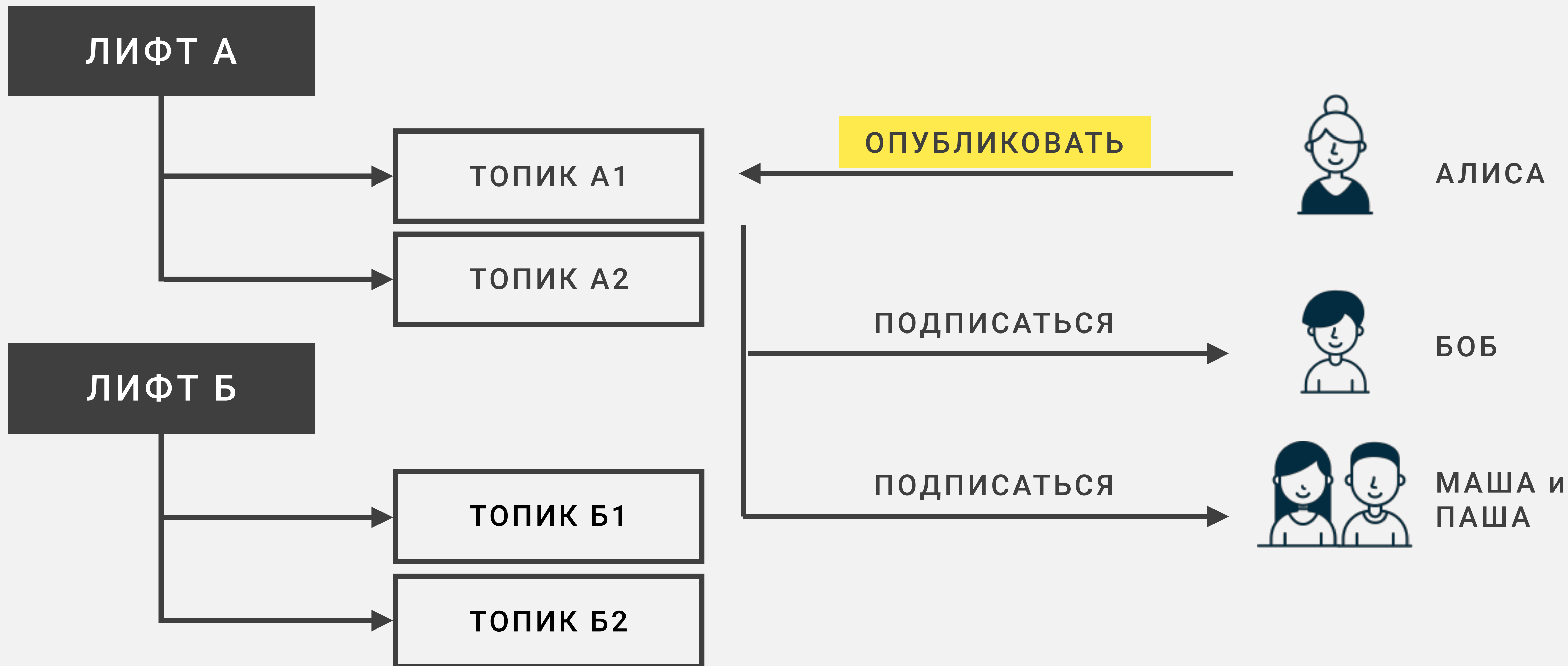


GET Send Save

Params Authorization Headers (2) Body Pre-request Script **Tests** Cookies Code

```
1 // Tests
2 try {
3   // response
4   pm.test("Response is JSON", function () {
5     pm.response.to.be.json;
6   });
7   // status
8   pm.test("Status code is 401", function () {
9     pm.response.to.have.status(401);
10  });
11  // error message
12  pm.test("Error data returned: Unauthorized", function () {
13    pm.expect(pm.response.json().message).to.equal("Unauthorized");
14  });
15 } catch(e) {
16   pm.test("Ugh, some checks could not complete!", function () { pm.expect.fail() });
17 } finally {}
18
```

MQTT: МОДЕЛЬ “ПУБЛИКАЦИЯ-ПОДПИСКА”



MQTT: ПРИМЕР

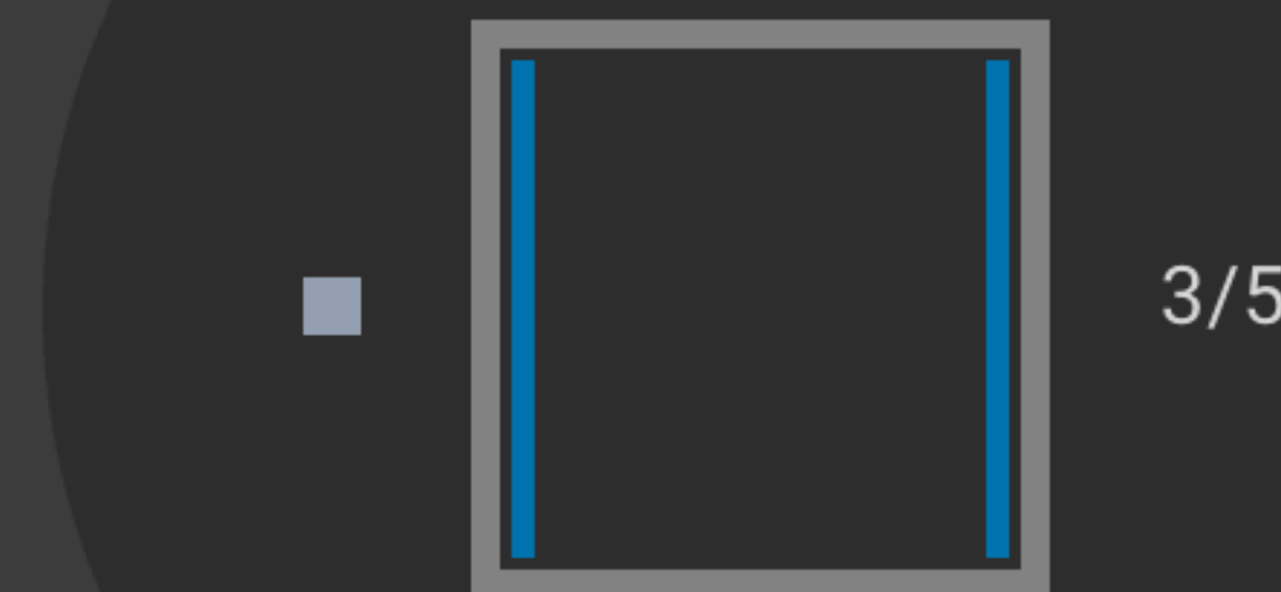


лифты/ID1234/состояние

```
{  
  "этаж": "3",  
  "направление": "стоп",  
  "двери": "открыты"  
}
```

ТОПИК

СООБЩЕНИЕ



СОБЫТИЯ И РЕАКЦИЯ



СОБЫТИЕ:

Таймер

Изменение состояния

Сообщение-запрос

СОБЫТИЯ И РЕАКЦИЯ



СОБЫТИЕ:

Таймер

Изменение состояния

Сообщение-запрос



РЕАКЦИЯ:

Сообщения в топике

Сообщения в топике

Сообщения в топике

СОБЫТИЯ И РЕАКЦИЯ



СОБЫТИЕ:

Сообщения в топике

Сообщения в топике

Сообщения в топике



РЕАКЦИЯ:

Запуск лямбды

SMS

Сообщение в топике

MQTT API: ТЕСТОВАЯ СТРАТЕГИЯ



1. Выявить события и реакцию на них
2. Воспроизвести позитивные и негативные случаи
3. Учесть тестирование производительности, безопасности, и другие аспекты
4. Адаптировать стратегию при необходимости

MQTT API ТЕСТЫ



1. Подписаться на топики

```
mosquitto_sub -t лифты/лифт123/состояние
```

2. Изменить состояние

3. Проверить сообщение в топике

```
{  
  "этаж": 4,  
  "двери": "закрываются"  
}
```

СРЕДСТВА ТЕСТИРОВАНИЯ



- AWS IoT Web Client
- AWS IoT REST API
- Eclipse paho
- **Eclipse mosquitto CLI**
- Mqtt.js
- MQTT.fx
- robot framework (MQTT Library)

СРЕДСТВА ТЕСТИРОВАНИЯ

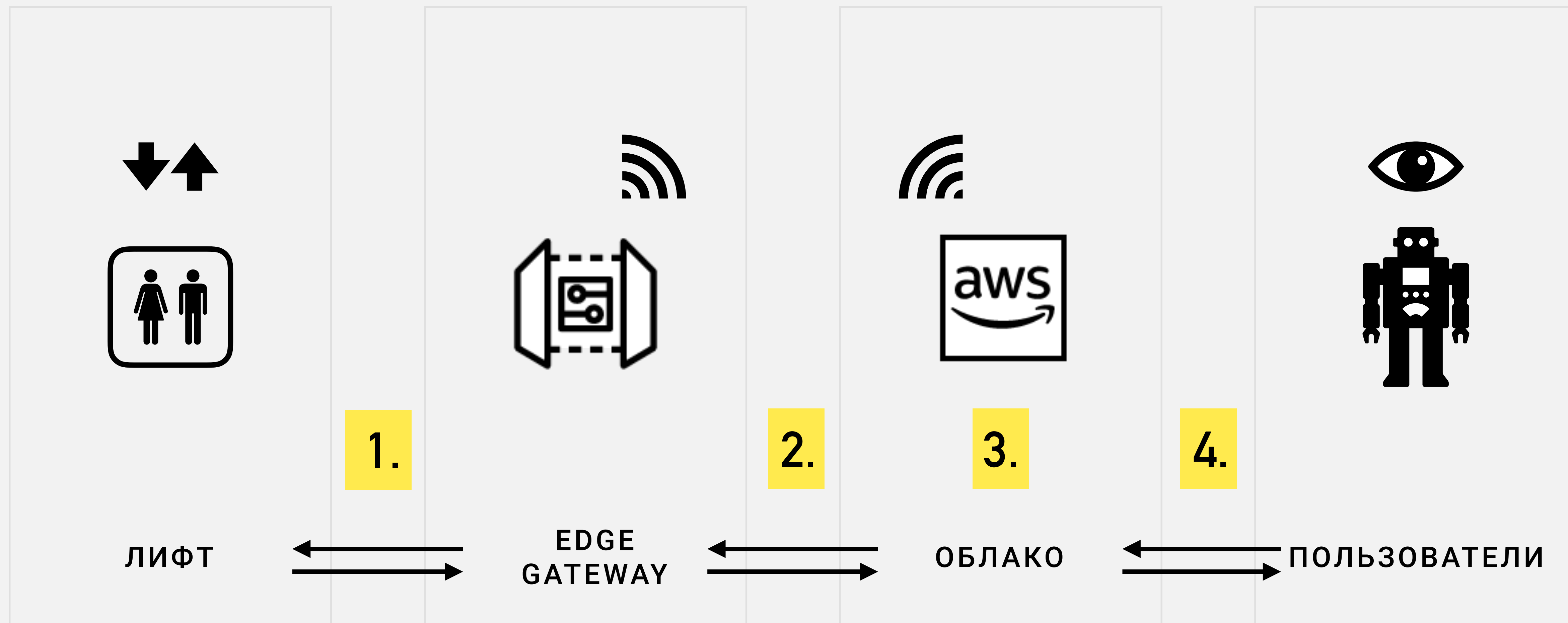


```
13 # send deviceMoved with floor = 1
14 ${LIFT_FLOOR}=1
15 echo '{
16   "liftId": "'${LIFT_ID}'",
17   "environment": "'${ENVIRONMENT_NAME}'",
18   "body": {
19     "ts": "'`date -u +%FT%TZ`'",
20     "name": "floorChanged",
21     "attributes": {
22       "floor": "'${LIFT_FLOOR}'",
23       "heightInCm": 13
24     }
25   }
26 }' | \
27 mosquitto_pub -s \
28 -h ${AWS_IOT_BROKER_ADDRESS}.amazonaws.com \
29 -p ${AWS_IOT_BROKER_PORT} \
30 -I ${TEST_CLIENT_ID_PUB} \
31 -t lifts/${LIFT_ID}/events \
32 -q 0 \
33 --tls-version tlsv1.2 \
34 --key ${TEST_CERT}.key \
35 --cert ${TEST_CERT}.pem \
36 --cafile ${TEST_CERT}.ca \
37 -d
```

СООБЩЕНИЕ

КОМАНДА
ПУБЛИКАЦИИ

IIOT ПЛАТФОРМА



ЗАЛОГ УСПЕХА

- 1. Разобраться в системе и технологиях**
- 2. Создать набор тестовых данных**

НЕВОШЕДШЕЕ: “ШАПОЧКА” ИЗ ФОЛЬГИ



НЕВОШЕДШЕЕ: ОБНОВЛЕНИЕ FIRMWARE



ДОП. МАТЕРИАЛЫ

1. <https://www.satisfice.com/download/heuristic-test-strategy-model>
2. <https://docs.aws.amazon.com/iot/latest/developerguide/what-is-aws-iot.html>
3. <http://www.steves-internet-guide.com/mqtt/>
4. <https://twitter.com/HeleneKolpakova>

СПАСИБО



#Heisenbug

ВОПРОСЫ

@HeleneKolpakova

elena.kolpakova@akqa.com

