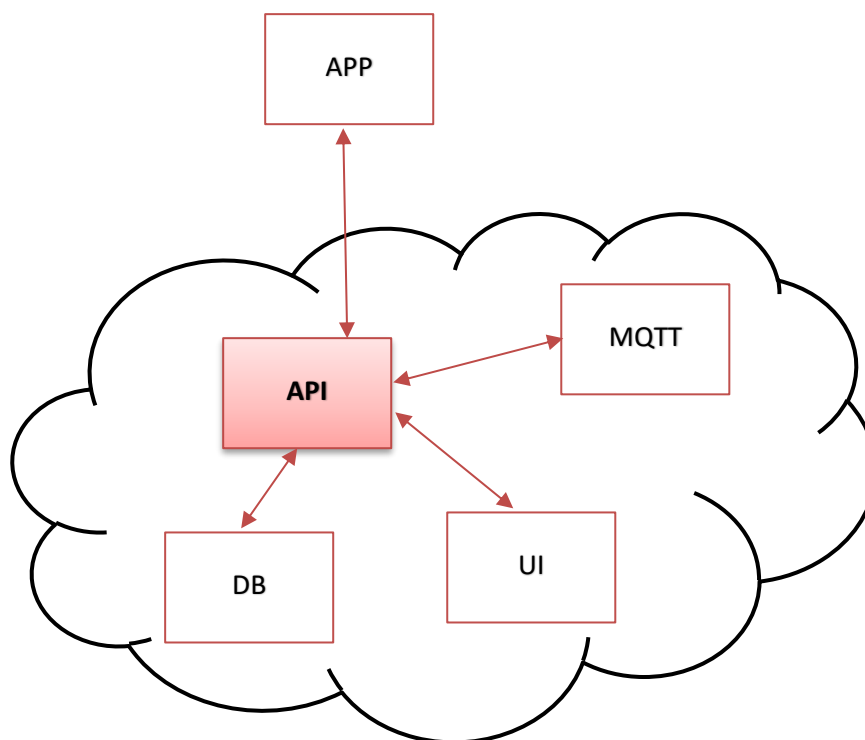


## API DOKUMENTACIJA

Primarna svrha aplikacijskog programskog sučelja je ponuda usluge međusobne komunikacije drugim dijelovima softwarea, u našem slučaju je to interakcija baze podataka, aplikacije, korisničkog sučelja i MQTTa. Prikaz međusobne komunikacije vidljiv je i na slici koja slijedi.



Implementacija će biti u C#/.NET okruženju koristeći standardne HTTP metode, mogući odgovori zahtjeva prikazani su u tablici koja slijedi u kojoj je kratko naznačeno i za što se koristi pojedina HTTP metoda.

HTTP metoda	CRUD	Uspješan odgovor	Neuspješan odgovor
POST	Kreirati	200 (OK)	404 (Bad Request)
GET	Čitati	200 (OK)	404 (Bad Request) 404 (Not Found)
PUT	Ažurirati	200 (OK)	404 (Not Found)
DELETE	Obrisati	200 (OK)	404 (Bad Request) 404 (Not Found)

Tablica 1: Opis HTTP metoda

Baza podataka sastoji se od pet tablica. Tablica 2 sadrži popis tablica u bazi podataka te njihov odnos s implementiranim radnjama nad njima. Znak kvačice (✓) označava da je radnja implementirana nad tom tablicom dok negaciju te tvrdnje predstavlja oznaka (✗).

Nakon tablice slijedi konkretan opis korištenja HTTP zahtjeva u aplikaciji popraćen primjerima.

Naziv tablice	Get all	Get one	Update	Delete	Add
Roles	✓	✓	✗	✓	✓
Login	✓	✓	✗	✓	✓
Devices	✓	✓	✓	✓	✓
Users	✓	✓	✓	✓	✓
Events	✓	✓	✓	✓	✓

## 1. HttpGet

HttpGet zahtjev šalje se prilikom dohvaćanja podataka iz baze, u konkretnom slučaju to se odnosi na:

- dohvaćanje svih podataka iz pojedine tablice
- dohvaćanje jednog određenog podatka definiranog po ID-u
- osiguravanje jedinstvene email adrese prilikom registracije
- dohvaćanje korisnika prilikom logina

### 1.1. Dohvaćanje svih zapisa tablice u bazi podataka

Prilikom slanja ovoga zahtjeva metodom GetAll potrebno je poslati ključnu riječ kojom se definira iz koje tablice se žele dohvatiti podatci.

Moguće ključne riječi:

- users
- devices
- events
- roles
- login

Zahtjev nije osjetljiv na velika i mala slova no ukoliko se upiše neka druga riječ zahtjev vraća 404 "Bad Request" odgovor.

Primjer zahtjeva za dohvaćanje svih zapisa iz tablice Devices:

<https://air2218.mobilisis.hr/api/api/VendingMachine/GetAll?ObjectType=devices>

### 1.2. Dohvaćanje određenog podatka iz tablice

Slanje ovoga zahtjeva podrazumijeva slanje id kojim će se identificirati zapis koji je potrebno dohvatiti te ključnu riječ kojoj se definira tablica u bazi podataka iz koje se dohvaća zapis, implementirano je u metodi GetOne.

Moguće ključne riječi:

- user
- device
- event
- role
- login

Zahtjev nije osjetljiv na velika i mala slova no ukoliko se upiše neka druga riječ zahtjev vraća 404 "Bad Request" odgovor. U slučaju kada se u tablici ne nalazi zapis sa zahtijevanim ID-om zahtjev vraća 404 "Not Found".

Primjer zahtjeva za dohvaćanje prvog zapisa iz tablice Devices:

<https://air2218.mobilisis.hr/api/api/VendingMachine/GetOne?id=1&ObjectType=device>

### 1.3. Osiguravanje jedinstvene mail adrese prilikom registracije

Koristeći metodu UniqueEmail šalje se email adresa kojoj se korisnik pokušava registrirati. Ukoliko korisnik s tom adresom postoji u bazi podataka odgovor je 404 "Bad Request" i pronađeni korisnik. U slučaju da ne postoji takav korisnik odgovor je 200 "OK".

Primjer zahtjeva za provjeru mail adrese primjer@primjer.hr:

<https://air2218.mobilisis.hr/api/api/VendingMachine/UniqueEmail?email=primjer%40primjer.hr>

### 1.4. Dohvaćanje korisnika prilikom logina

Ovim zahtjevom šalje se mail adresa i lozinka kojom se korisnik želi prijaviti. Ukoliko u bazi ne postoji korisnik koji ima identične podatke vraća s odgovor 404 "Not Found", inače je odgovor 200 "OK" i pronađeni korisnik.

Primjer zahtjeva s emailom primjer@primjer.hr i lozinkom primjer :

<https://air2218.mobilisis.hr/api/VendingMachine/LoginRequest?email=primjer%40primjer.hr&password=primjer>

## 2. HttpPost

Ovaj zahtjev šalje se u slučaju dodavanja zapisa u bazu podataka. Moguće je poslati pet različitih vrsta post zahtjeva, za svaku tablicu po jedan. Prilikom slanja zahtjeva za dodavanje u tablicu Roles šalje se samo string description dok se kod dodavanja u tablicu Login šalje Integer user\_id i DateTime date\_time. Kod slanja zahtjeva za dodavanje u ostalim tablicama koristi se određena request body shema koja odgovara tipu podatka tablice u koju se dodaje.

- Shema za tablicu Devices

```
{
  "lat": 0,
  "long": 0,
  "stock": 0,
  "price": 0,
  "active": true
}
```

- Shema za tablicu Users

```
{
  "first_name": "string",
}
```

- ```

    "last_name": "string",
    "email": "string",
    "password": "string",
    "role_id": 0
  }

```
- Shema za tablicu Events
 

```

      {
        "user_id": 0,
        "device_id": 0,
        "date_time": "2023-03-06T10:54:50.537Z"
      }
      
```

**Primjer zahtjeva za dodavanje novog zapisa u tablicu Devices:**

zahtjev:

<https://air2218.mobilisis.hr/api/api/VendingMachine/AddDevice>

body:

```

{
  "lat": 46.31,
  "long": 16.36,
  "stock": 78,
  "price": 1.10,
  "active": true
}

```

headres:

| KEY          | VALUE            |
|--------------|------------------|
| Content-Type | application/json |

### 3. HttpPut

Http zahtjev Put šalje se prilikom ažuriranja podatka u bazi, a moguć je za tri tablice, Devices, Users i Events. Zahtjev za ažuriranje zapisa sastoji se od id-a kojim se identificira zapisa za ažuriranje te request body sheme koja je specifična za svaku tablicu:

- Shema za tablicu Devices
 

```

      {
        "lat": 0,

```

```
"long": 0,  
"stock": 0,  
"price": 0,  
"active": true  
}
```

- Shema za tablicu Users

```
{  
  "first_name": "string",  
  "last_name": "string",  
  "email": "string",  
  "password": "string",  
  "role_id": 0  
}
```

- Shema za tablicu Events

```
{  
  "user_id": 0,  
  "device_id": 0,  
  "date_time": "2023-03-06T10:54:50.537Z"  
}
```

**Primjer zahtjeva za ažuriranje zapisa kojemu je id jednak 15 u tablici Devices:**  
zahtjev:

<https://air2218.mobilisis.hr/api/api/VendingMachine/UpdateDevice?id=15>

body:

```
{  
  
  "lat": 46.31,  
  
  "long": 16.36,  
  
  "stock": 78,  
  
  "price": 2.10,  
  
  "active": false  
}
```

headres:

| KEY          | VALUE            |
|--------------|------------------|
| Content-Type | application/json |

#### 4. **HttpDelete**

Kao što i sama riječ kaže, ova metoda omogućava brisanje zapisa iz baze. Za uspješno brisanje potrebno je poslati id zapisa i ključnu riječ kojom će se odrediti iz koje tablice je potrebno obrisati zapis.

Moguće ključne riječi:

- user
- device
- event
- role
- login

Zahtjev nije osjetljiv na velika i mala slova no ukoliko se upiše neka druga riječ zahtjev vraća 404 "Bad Request" odgovor. U slučaju kada se u tablici nalazi zapis sa zahtijevanim ID-om zahtjev vraća 200 "OK" i obrisani objekt. Ukoliko se zapis sa zadanom identifikacijom ne nalazi u odabranoj tablici odgovor je 404 "Not Found".

**Primjer zahtjeva za brisanje zapisa u tablici Devices kojemu je id jednak 15:**

<https://air2218.mobilisis.hr/api/api/VendingMachine?id=15&ForDelete=device>