

Ujian ini bertujuan untuk mengetahui keterampilan dasar kandidat dalam membuat solusi dari masalah yang ada dengan tidak mengabaikan standar yang ditentukan.

## Tugas

Buat sebuah aplikasi yang berjalan di server / back-end, dengan mengusung konsep Rest-API.

## Syarat & Ketentuan

1. Menggunakan Symfony 4 (terbaru).
2. Menggunakan Doctrine ORM 2 (terbaru).
3. Kode dipublikasi ke Github.

Sebisa mungkin terapkan gaya pengkodean yang direkomendasikan oleh Symfony.

## Kebutuhan Teknis

### Entitas

#### Pasien

- **id** - number autoincrement required
- **no\_rm** - string(8) unique autogenerate required (*contoh: 00000001, 00000002 ...dst*)
- **nama** - string(255) required
- **jenis\_kelamin** - string(20) required
- **tanggal\_lahir** - date required

#### Kunjungan

- **id** - number autoincrement required
- **pasien** - Pasien(many to one) required
- **tanggal\_masuk** - datetime autogenerate required
- **tanggal\_keluar** - datetime

### Permintaan

#### GET /pasien

Mengambil semua data pasien

*Response*

- 200 - semua data pasien

```
[
  {
    "id": 1,
    "no_rm": "string",
    "nama": "string",
    "jenis_kelamin": "string",
    "tanggal_lahir": "string"
  }
]
```

## GET /pasien/{id<\d+>}

Mengambil data pasien berdasarkan `id`

*Response*

- 200 - data pasien dengan `id = 1`

```
{
  "id": 1,
  "no_rm": "string",
  "nama": "string",
  "jenis_kelamin": "string",
  "tanggal_lahir": "string"
}
```

- 404 - data pasien tidak ditemukan

## POST /pasien

Membuat data pasien

*Body*

```
{
  "nama": "string",
  "jenis_kelamin": "string",
  "tanggal_lahir": "string"
}
```

*Response*

- 201 - data pasien baru

```
{
  "id": 1,
  "no_rm": "string",
  "nama": "string",
  "jenis_kelamin": "string",
  "tanggal_lahir": "string"
}
```

- 400 - masukan tidak valid

## PUT /pasien/{id<\d+>}

Mengubah data pasien dengan `id = 1`

*Body*

```
{
  "nama": "string",
  "jenis_kelamin": "string",
  "tanggal_lahir": "string"
}
```

*Response*

- 200 - data pasien setelah diubah

```
{
  "id": 1,
  "no_rm": "string",
  "nama": "string",
  "jenis_kelamin": "string",
  "tanggal_lahir": "string"
}
```

- 400 - masukan tidak valid
- 404 - data pasien tidak ditemukan

## DELETE /pasien/{id<\d+>}

Menghapus data pasien dengan `id = 1`

*Response*

- 204 - data pasien berhasil dihapus
- 404 - data pasien tidak ditemukan

## GET /kunjungan

Mengambil semua data kunjungan.

*Rule*

- `tanggal_keluar is null`

*Response*

- 200 - semua data kunjungan

```
[
  {
    "id": 1,
    "pasien": {
      "id": 1,
      "no_rm": "string",
      "nama": "string",
      "jenis_kelamin": "string",
      "tanggal_lahir": "string"
    },
    "tanggal_masuk": "string",
    "tanggal_keluar": null
  }
]
```

## GET /kunjungan/pasien/{pasienId<\d+>}

Mengambil semua data kunjungan berdasarkan `pasien_id = {pasienId}`.

*Rule*

- `tanggal_keluar is null`

*Response*

- 200 - semua data kunjungan

```
[
  {
    "id": 1,
    "pasien": {
      "id": 1,
      "no_rm": "string",
      "nama": "string",
      "jenis_kelamin": "string",
      "tanggal_lahir": "string"
    },
    "tanggal_masuk": "string",
    "tanggal_keluar": null
  }
]
```

- 404 - data pasien tidak ditemukan

## POST /kunjungan

Membuat data kunjungan

Body

```
{
  "pasien_id": 1
}
```

Response

- 201 - data kunjungan baru

```
{
  "id": 1,
  "pasien": {
    "id": 1,
    "no_rm": "string",
    "nama": "string",
    "jenis_kelamin": "string",
    "tanggal_lahir": "string"
  },
  "tanggal_masuk": "string",
  "tanggal_keluar": null
}
```

- 400 - masukan tidak valid
- 404 - data pasien tidak ditemukan

## PUT /kunjungan/{id<d+>}/pulang

Mengubah status data kunjungan dengan `id = 1` dari aktif ke pulang, atau dari `tanggal_keluar = null` ke `tanggal_keluar = NOW`.

Rule

- `tanggal_keluar is null`

Response

- 200 - data kunjungan setelah diubah

```
{
  "id": 1,
  "pasien": {
    "id": 1,
    "no_rm": "string",
    "nama": "string",
    "jenis_kelamin": "string",
    "tanggal_lahir": "string"
  },
  "tanggal_masuk": "string",
  "tanggal_keluar": "string"
}
```

- 404 - data kunjungan tidak ditemukan

## DELETE /kunjungan/{id<d+>}

Menghapus data kunjungan dengan `id = 1`.

Rule

- `tanggal_keluar is null`

Response

- 204 - data kunjungan berhasil dihapus
- 404 - data kunjungan tidak ditemukan