

Documentation

Campaign Voucher for Backend

- **API Documentation**
- **API Flowchart**
- **Usage steps Application**
- **Diagram Database**

API Documentation:

Running on port: 8080

Base URL: localhost:8080

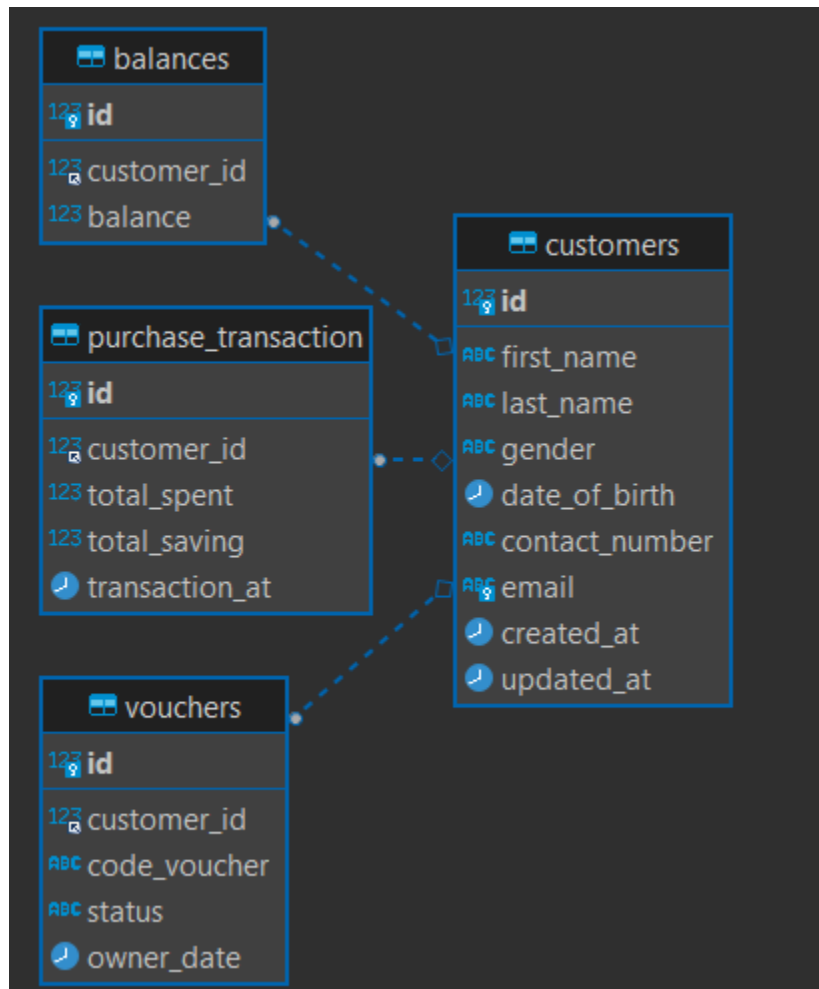
List API:

1. **Customer eligible check:**
 - Endpoint: **/api/campaign/checking**
2. **Validate photo submission:**
 - Endpoint: **/api/campaign/validate-profile**
3. **API for create customer:**
 - Endpoint: **/api/customer/add**
4. **API for create balance / topup:**
 - Endpoint: **/api/balance/topup**
5. **API for create transaction:**
 - Endpoint: **/api/transaction/add**
6. **API for create voucher:**
 - Endpoint: **/api/voucher/add**

Usage steps:

1. Hit query sql for generate Database
2. Hit API for create customer
3. Hit API for create balance (topup)
4. Hit API for create transaction
5. Hit API for create voucher
6. Hit API for customer eligible check
7. Hit API for validate photo submission

Design Database:



Customer eligible check:

Endpoint: **/api/campaign/checking**

Method: **POST**

Request Body:

- **Data Body:**
 - o **Email:** string

```
{
  "email": "tes@user.com"
}
```

Response:

- **Success**
 - o **Response code:** 200

```
{
  "status": 200,
  "message": "You are entitled to a voucher!",
  "data": {
    "total_transaction": 5,
    "total_transaction_amount": "198.00",
    "duration": "2022-9-17 00:00:00 until 2022-10-17 23:59:59",
    "campaign_eligibility": true,
    "time_expired": "10/17/2022, 1:02:50 PM"
  }
}
```

- **Failed (Not found)**
 - o **Response code:** 404

```
{
  "status": 404,
  "message": "Customer not found.",
  "data": null
}
```

Validate photo submission:

Endpoint: **/api/campaign/validate-profile**

Method: **POST**

Request Body:

- **Data Body:**
 - o **Email:** string
 - o **img_profile:** boolean

```
{
  "email": "tes@user.com",
  "img_profile": true
}
```

Response:

- **Success**
 - o **Response code:** 200

```
{
  "status": 200,
  "message": "Yeah, you get a voucher!",
  "data": {
    "user_id": 1,
    "email": "tes@user.com",
    "campaign_eligibility": true,
    "code_voucher": "a-333",
    "status_voucher": "closed",
    "is_expired": false
  }
}
```

- **invalid**
 - o **Response code:** 201

```
{
  "status": 201,
  "message": "Validation image profile not valid",
  "data": {
    "user_id": 1,
    "email": "tes@user.com",
    "campaign_eligibility": false
  }
}
```

API for create customer:

Endpoint: **/api/customer/add**

Method: **POST**

Request Body:

- **Data Body:**
 - o **All parameter:** string

```
{
  "first_name": "Yoko",
  "last_name": "Hama",
  "gender": "Male",
  "date_of_birth": "1995-05-01",
  "contact_number": "08787878787",
  "email": "tes@user.com"
}
```

Response:

- **Success create user**
 - o **Response code:** 200

```
{
  "status": 200,
  "message": "Success created"
}
```

- **Failed**
 - o **Response code:** 500

```
{
  "status": 500,
  "message": "email must be unique"
}
```

API for create balance / topup:

Endpoint: **/api/balance/topup**

Method: **POST**

Request Body:

- **Data Body:**
 - o **email:** string
 - o **topup:** integer

```
{  
  "email": "oke@oke.com",  
  "topup": 600  
}
```

Response:

- **Success create balance**
 - o **Response code:** 200

```
{  
  "status": 200,  
  "message": "Success created topup balance.",  
  "total_balance": 600  
}
```

- **Failed**
 - o **Response code:** 500

```
{  
  "status": 500,  
  "message": "Internal server error"  
}
```

API for create transaction:

Endpoint: **/api/transaction/add**

Method: **POST**

Request Body:

- **Data Body:**
 - o **email:** string
 - o **total_spent:** integer

```
{  
  "email": "tes@user.com",  
  "total_spent": 50  
}
```

Response:

- **Success create transaction**
 - o **Response code:** 200

```
{  
  "status": 200,  
  "message": "Success created transaction",  
  "data": {  
    "balance": 150  
  }  
}
```

- **Failed**
 - o **Response code:** 500

```
{  
  "status": 500,  
  "message": "Internal server error"  
}
```


API for create vouchers:

Endpoint: **/api/voucher/add**

Method: **POST**

Request Body:

- **Data Body:**
 - o **add_voucher:** integer

```
{  
  "add_voucher": 10  
}
```

Response:

- **Success create voucher**
 - o **Response code:** 200

```
{  
  "status": 200,  
  "message": "Success create 10 vouchers.",  
  "data": {  
    "total_voucher": 10  
  }  
}
```

- **Failed**
 - o **Response code:** 400

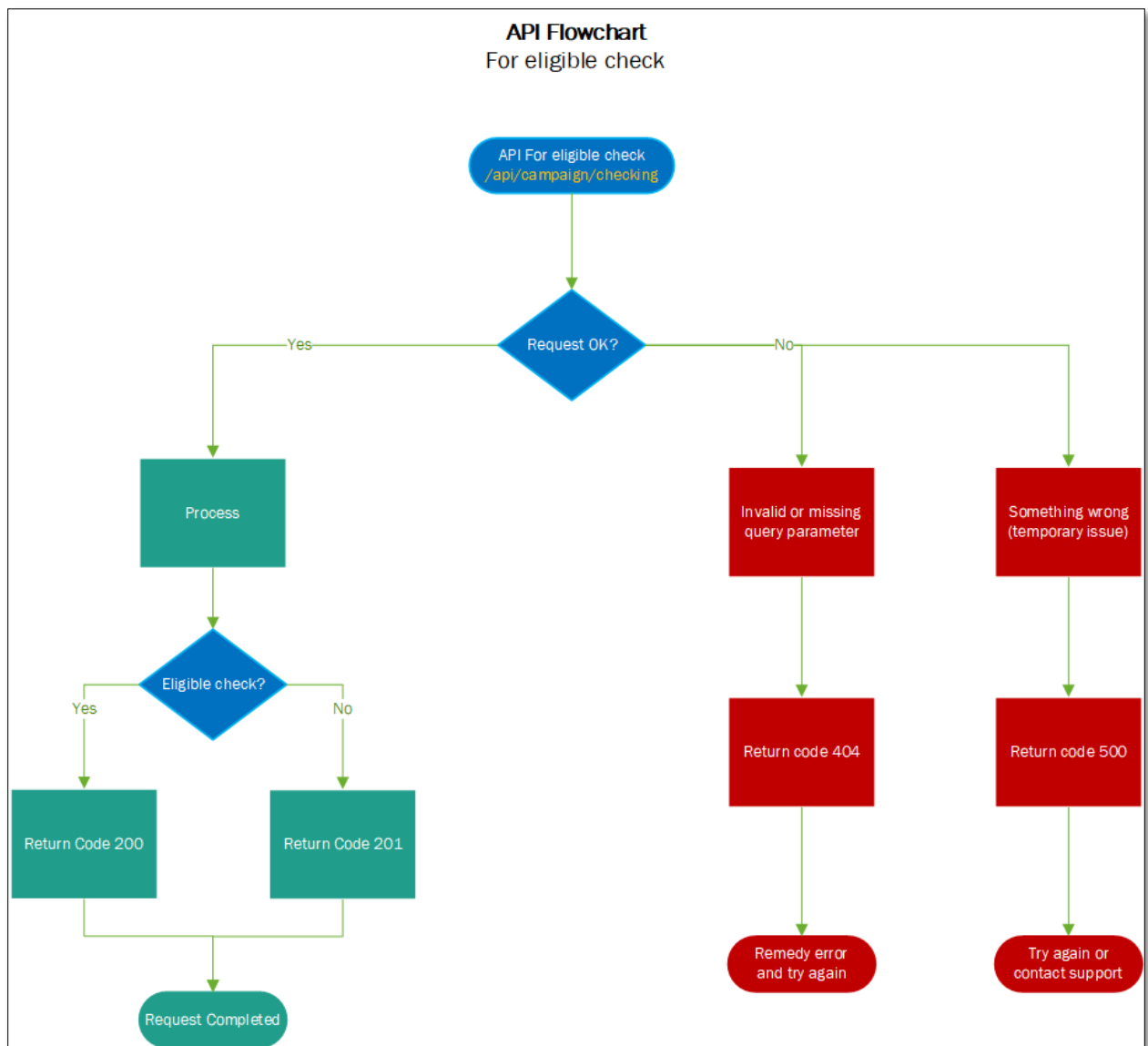
```
{  
  "status": 400,  
  "message": "Content can not be empty!"  
}
```

- **Failed**
 - o **Response code:** 500

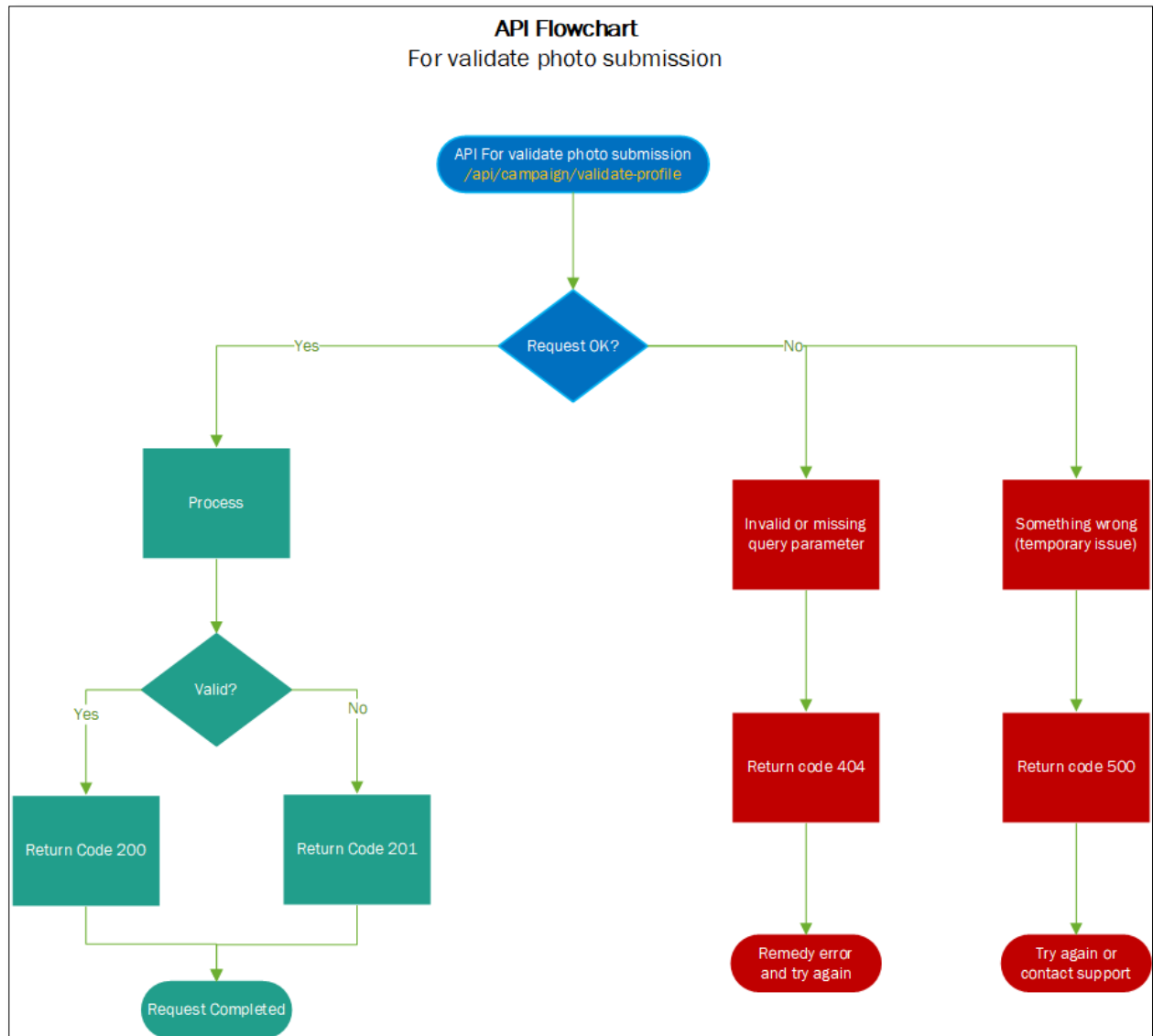
```
{  
  "status": 500,  
  "message": "Internal server error"  
}
```

API Flowchart

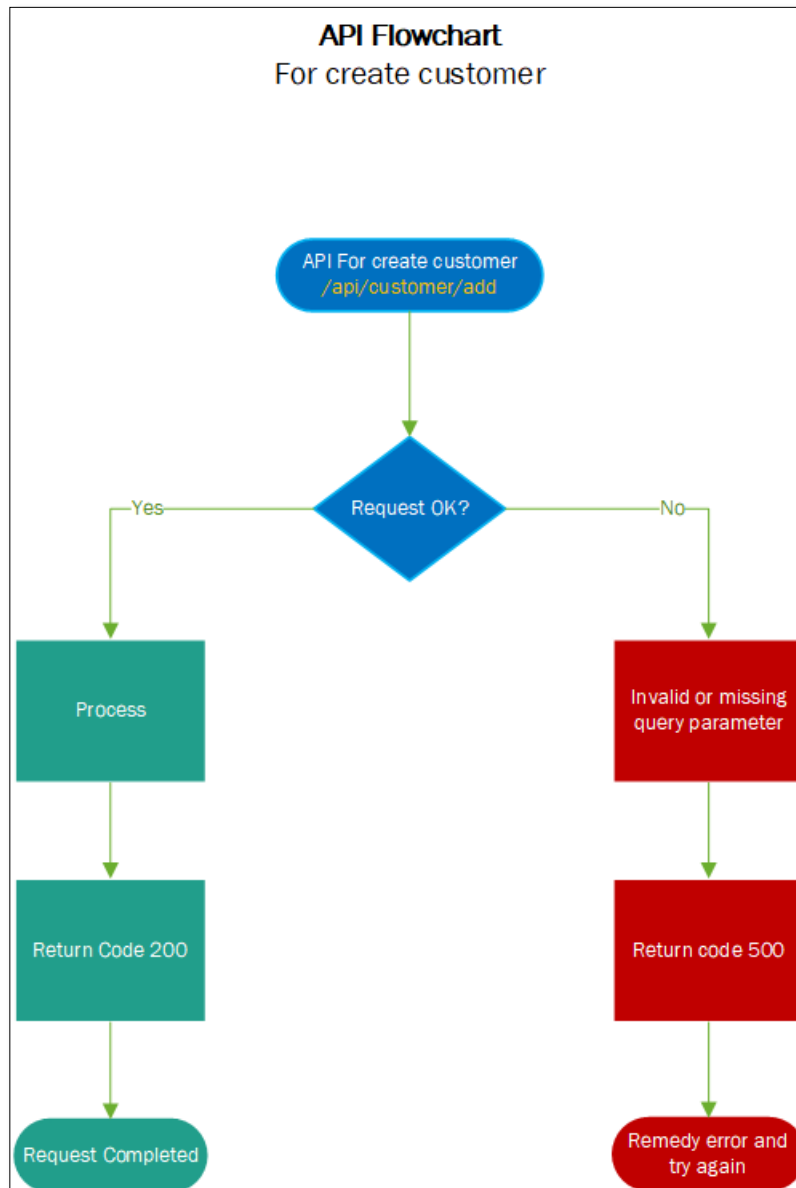
Customer eligible check:



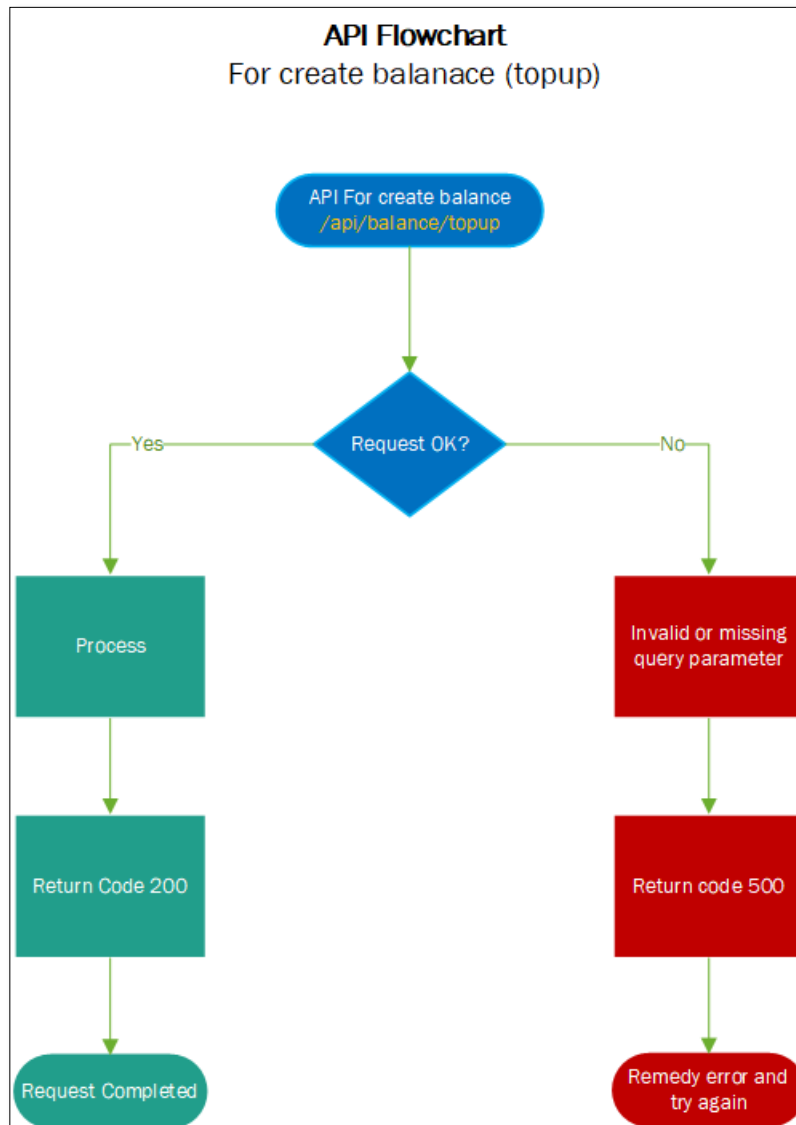
Validate photo submission:



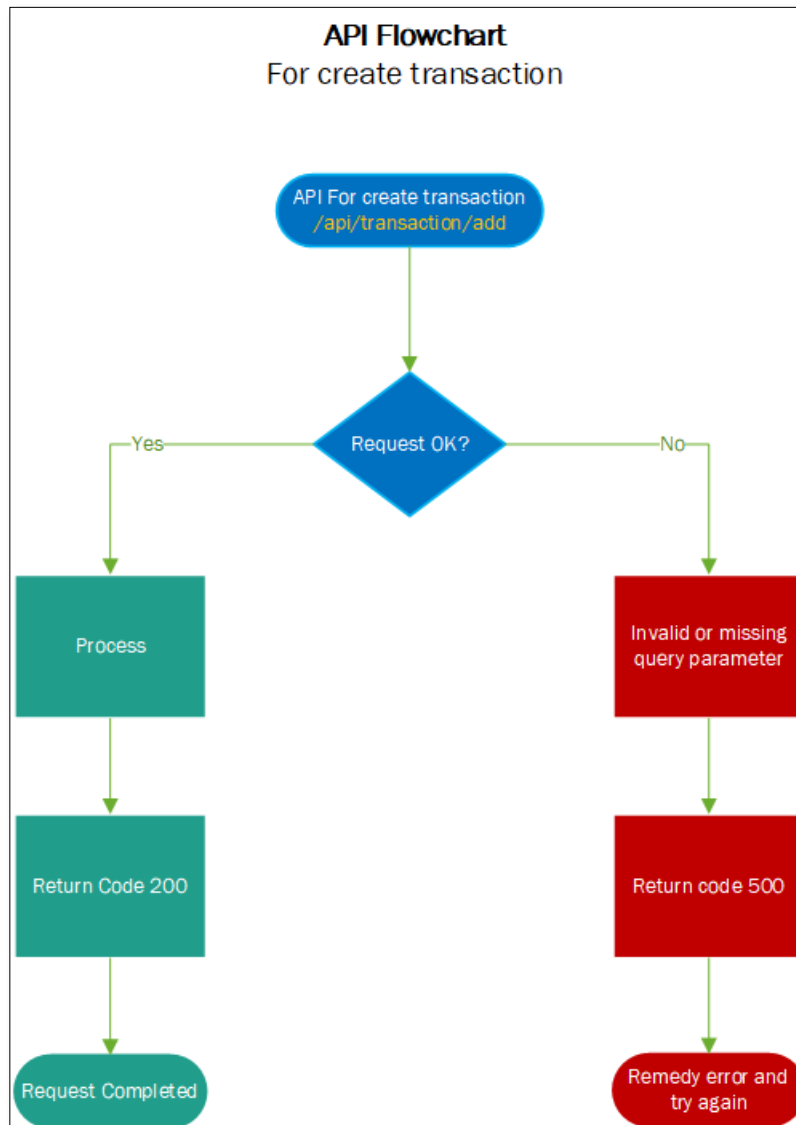
API for create customer:



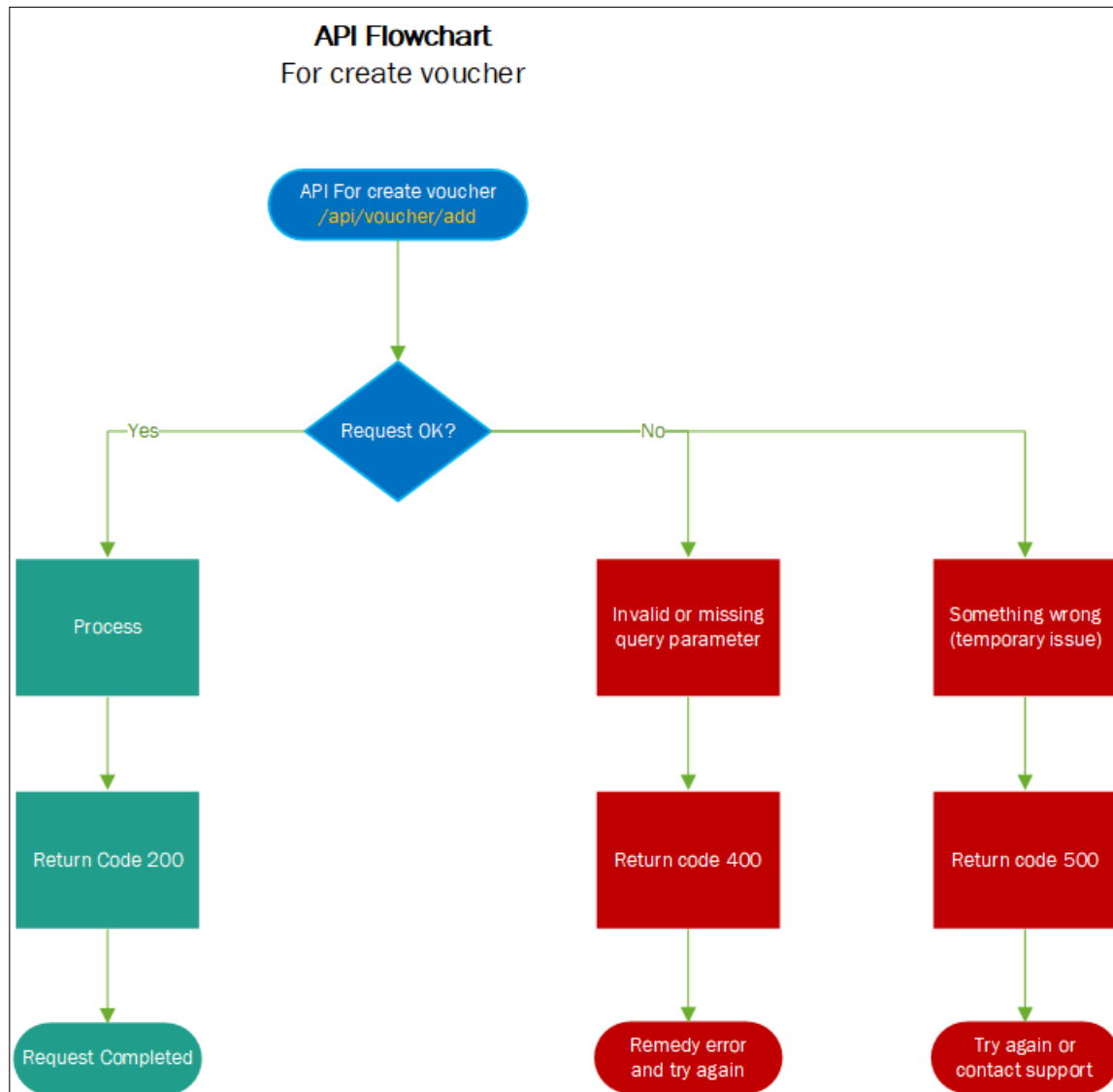
API for create balance:



API for create transaction:



API for create voucher:



Technology:

Framework:

- Node JS for Backend

Run Application:

- npm install
- npm run start

ORM:

- Sequelize

DB Manager:

- MySQL

Packages:

- cors
- express
- mysql2
- sequelize
- nodemon