

# 1 Opis API

Dokumentacja API modułu bazy przygotowana została przy użyciu specyfikacji OpenAPI 2.0.0.

## 1.1 Rejestracja, logowanie, banowanie i obsługa kont klienckich

Dokumentacja w sposób celowy nie zawiera opisu autoryzacji, przechowywania informacji o użytkowniku, zmiany hasła, banowania użytkowników. Wszystkie te elementy będą zrealizowane za pomocą Azure Active Directory B2C (zwanego dalej AAD) i zawartego tam mechanizmu opcjonalnych claims'ów. Po zalogowaniu się do portalu AAD do zapytań wysyłanych w kierunku serwera dołączany będzie token otrzymany z AAD, który będzie zawierał potrzebne informacje na temat użytkownika i który (w uproszczeniu, gdyż AAD dołącza również pola na swoje własne potrzeby) będzie miał następującą postać:

Token:

```
type: "object"
properties:
  oid:
    type: "string"
    description: "unique guid, used to disctinct registered users"
  userType:
    $ref: "#/definitions/UserType"
  email:
    type: "string"
  name:
    type: "string"
  surname:
    type: "string"
  address:
    $ref: "#/definitions/Address"
  isBanned:
    type: "boolean"
```

Dokładny opis klas zawartych w tokenie został podany niżej razem z opisem całego api. Z racji na to, że usługa Azure Active Directory B2C jest cały czas rozwijana, modyfikowana i ulepszana opis api nie zawiera dokładnego opisu wiadomości wysyłanych z naszej aplikacji do AAD. Dodatkowo wiadomości te będą wysyłane za pomocą zewnętrznego api (opisanego dokładniej w rozdziale Technologia), zatem z punktu widzenia rozwijanego oprogramowania dokładny opis tych wiadomości oraz struktur nie jest istotny.

Jak widać wszystkie "stałe" informacje będą zawarte w tokenie pobranym z AAD, natomiast informacje które często mogą ulegać zmianie jak grafik, filtry itp. zostały opisane niżej w dokładnej dokumentacji API. również usuwanie klienta, ze względu na ograniczenia AAD, będzie realizowany w standardowy sposób poprzez wysłanie zapytania na moduł serwera.

## 1.2 Dokładna dokumentacja API

Dla wszystkich poniższych requestów zakładamy że mają one postać: [nazwa aplikacji]/api/[reszta url]

### 1.2.1 Obsługa zamówień

```
/orders:
  get:
    summary: "Get all orders"
    produces:
```

```

    - "application/json"
  responses:
    "200":
      description: "OK"
      schema:
        type: "array"
        items:
          $ref: "#/definitions/Order"
    "403":
      description: "Not Authorised"
  post:
    summary: "Add order"
    consumes:
      - "application/json"
    parameters:
      - name: "body"
        in: body
        schema:
          $ref: "#/definitions/NewOrder"
    responses:
      "200":
        description: "OK"
      "403":
        description: "Not Authorised"
      "400":
        description: "Bad Request"
/orders/{orderId}:
  get:
    summary: "Get order info"
    produces:
      - "application/json"
    responses:
      "200":
        description: "OK"
        schema:
          $ref: "#/definitions/Order"
      "403":
        description: "Not Authorised"
  post:
    summary: "Modify order info"
    consumes:
      - "application/json"
    parameters:
      - name: "body"
        in: body
        schema:
          $ref: "#/definitions/Order"
    responses:
      "200":
        description: "OK"
      "403":

```

```

        description: "Not Authorised"
    "400":
        description: "Bad Request"
delete:
    summary: "Removes offer"
    consumes:
        - "application/json"
    responses:
        "200":
            description: "OK"
        "403":
            description: "Not Authorised"

```

### 1.2.2 Operacje konta osoby sprzątającej

```

/cleaner/{cleanerId}:
get:
    summary: "retrieves cleaner information including filters and
        schedule"
    consumes:
        - "application/json"
    responses:
        "200":
            description: "OK"
            schema:
                $ref: "#/definitions/CleanerInfo"
        "403":
            description: "Not Authorised"
post:
    summary: "updates cleaner information"
    consumes:
        - "application/json"
    parameters:
        - name: "body"
          in: body
          schema:
              $ref: "#/definitions/CleanerInfo"
    responses:
        "200":
            description: "OK"
        "403":
            description: "Not Authorised"
        "400":
            description: "Bad Request"
/cleaner/{cleanerId}/orders:
get:
    summary: "Get assigned orders"
    produces:
        - "application/json"
    responses:
        "200":

```

```

    description: "OK"
  schema:
    type: "array"
    items:
      $ref: "#/definitions/Order"
  "403":
    description: "Not Authorised"

```

### 1.2.3 Operacje dla kont użytkowników(klienta i osoby sprzątającej)

```

/client/{id}:
  delete:
    summary: "Deletes account from the system"
    produces:
      - "application/json"
    responses:
      "200":
        description: "OK"
      "403":
        description: "Not Authorised"
/user/{id}/rate:
  post:
    summary: "rates client/cleaner by opposite side (in connection to
      execution of an order)"
    produces:
      - "application/json"
    parameters:
      - name: "body"
        in: body
        schema:
          $ref: "#/definitions/AddRating"
    responses:
      "200":
        description: "OK"
      "403":
        description: "Not Authorised"
      "400":
        description: "Bad Request"
/user/{id}/ban:
  post:
    summary: "ban client/cleaner"
    produces:
      - "application/json"
    responses:
      "200":
        description: "OK"
      "403":
        description: "Not Authorised"

```

#### 1.2.4 Ustawianie Prowizji

```
/Configuration/Commission:
  get:
    summary: "Get commission value"
    produces:
      - "application/json"
    responses:
      "200":
        description: "OK"
        schema:
          $ref: "#/definitions/SetCommission"
  post:
    summary: "sets commision for the orders"
    produces:
      - "application/json"
    parameters:
      - name: "body"
        in: body
        schema:
          $ref: "#/definitions/SetCommission"
    responses:
      "200":
        description: "OK"
      "403":
        description: "Not Authorised"
      "400":
        description: "Bad Request"
```

#### 1.2.5 Definicje Klas

```
definitions:
  MessLevel:
    type: "string"
    enum:
      - "Low"
      - "Moderate"
      - "Huge"
      - "Disaster"
  OrderStatus:
    type: "string"
    enum:
      - "Active"
      - "InProgress"
      - "Cancelled"
      - "Closed"
  Order:
    type: "object"
    properties:
      id:
```

```

        type: "integer"
        format: "int64"
    clientId:
        type: "string"
    cleanerId:
        type: "string"
    status:
        $ref: "#/definitions/OrderStatus"
    maxPrice:
        type: "number"
    minRating:
        type: "integer"
    messLevel:
        $ref: "#/definitions/MessLevel"
NewOrder:
    type: "object"
    properties:
        clientId:
            type: "string"
        maxPrice:
            type: "number"
        minRating:
            type: "integer"
        messLevel:
            $ref: "#/definitions/MessLevel"
        date:
            type: "date"
        address:
            $ref: "#/definitions/Address"
UserType:
    type: "string"
    enum:
        - "Client"
        - "Cleaner"
        - "Administrator"
Address:
    type: "object"
    properties:
        street:
            type: "string"
        buildingNo:
            type: "string"
        flatNo:
            type: "integer"
            nullable: true
        city:
            type: "string"
        postalCode:
            type: "integer"
        country:
            type: "string"

```

Token:

- type: "object"
- description: "fields that are included in the b2c claims that will be necessary to process requests"
- properties:
  - oid:
    - type: "string"
    - description: "unique guid, used to distinct registered users"
  - userType:
    - \$ref: "#/definitions/UserType"
  - email:
    - type: "string"
  - name:
    - type: "string"
  - surname:
    - type: "string"
  - address:
    - \$ref: "#/definitions/Address"
  - isBanned:
    - type: "boolean"

DayOfWeek:

- type: "string"
- enum:
  - "Monday"
  - "Tuesday"
  - "Wednesday"
  - "Thursday"
  - "Friday"
  - "Saturday"
  - "Sunday"

ScheduleEntry:

- type: "object"
- description: "single entry of Schedule"
- properties:
  - dayOfWeek:
    - \$ref: "#/definitions/DayOfWeek"
  - start:
    - type: "string"
    - description: "Time in format HH:MM from which availability starts"
  - end:
    - type: "string"
    - description: "Time in format HH:MM on which availability ends"

CleanerInfo:

- type: "object"
- description: "cleaner information necessary to match orders"
- properties:
  - scheduleEntries:
    - type: "array"
    - items:
      - \$ref: "#/definitions/ScheduleEntry"
  - maxMess:

```

    $ref: "#/definitions/MessLevel"
  minClientRating:
    type: "integer"
  minPrice:
    type: "number"
    format: "float"
  maxLocationRange:
    type: "number"
    format: "float"
SetCommission:
  type: "object"
  properties:
    newProvision:
      description: "new commision in fraction"
      type: "number"
      format: "decimal"
      maximum: 1.0
      minimum: 0.0
AddRating:
  type: "object"
  properties:
    rating:
      type: "integer"
      maximum: 10
      minimum: 1

```