МИНИСТЕРСТВО НА ТРАНСПОРТА И СЪОБЩЕНИЯТА

ИЗПЪЛНИТЕЛНА АГЕНЦИЯ „АВТОМОБИЛНА АДМИНИСТРАЦИЯ“

Ниво на класификация [TLP-WHITE]

ЗАПОВЕД

№ ……………… / ………. 2023 г.

На основание чл. 7, т. 1 и т. 2 от Устройствения правилник на Изпълнителна агенция „Автомобилна администрация“ и чл. 9, ал. 9б от Наредба № Н-32 от 16.12.2011 г. за периодичните прегледи за проверка на техническата изправност на пътните превозни средства (Наредба № Н-32)

НАРЕЖДАМ:

1. Определям съдържанието на комуникационния протокол, който трябва да имат имплементиран софтуерите, осигуряващи електронно предаване на резултатите от измерванията от газоанализаторите, димомерите и стендовете за измерване на спирачните сили на пътните превозни средства към информационната система за електронно регистриране на извършените периодични прегледи за проверка на техническата изправност на пътните превозни средства по чл. 11, ал. 3 от Наредба № Н-32 при извършване на периодичните прегледи за проверка на техническата изправност на пътните превозни средства, както следва:

Комуникационен протокол

I. Предаване на данни от газоанализатора

1. Description

The provided REST service transmits measurement data from an emission gas analyzer to an Executive Agency "Automobile Administration"'s client. The emission gas analyzer sends a request with the measured data and the EA "AA"'s client responds with a http status code.

2. Service endpoints

A request to this endpoint **must contain** the complete set of measured data.

```
Request method: POST

Request URL: /api/inspection/current/measurement/gas-emissions
```

3. Request

The request consists of the following data:

- `location` - Text - The location address where the measurement was conducted.
- `conductedDatetime` - Date in ISO 8601 format - The end date time of the measurement.
- `manufacturer` - Text - The measuring device's manufacturer name or brand name.

- `model` - Text - The measuring device's model.
- `serialNumber` - Text - The measuring device's serial number.
- `softwareVersion` - Text - The measuring device's software version. (optional)
- `deviceRemark` - Text - Additional remarks for the measuring device. (optional)
- `regNum` - Text in latin letters and nimbers - The vehicle's registration number.
- `measurements[]` - An array of performed inspections for each test. The first index of the array corresponds to test 1, the second to test 2. (Length 4)
  An element of the array represents an object with the following structure:
  - `coPercent` - Real number - Percentage of CO.
  - `co2Percent` - Real number - Percentage of CO2.
  - `o2Percent` - Real number - Percentage of O2.
  - `hc` - Real number - PPM of HC.
  - `airRatioLambda` - Real number - Air ratio values λ lambda. (Optional)
  - `rpmResult` - Number - Revolution per minute.
  - `engineOilTemperature` - Number - Engine oil temperature.
  - `fuelType` - Number - Fuel Type.

```
○    1 - Petrol

○    2 - LPG

○    3 - CNG
```

4. Response

The response of the EA "AA"'s client represents an http status code. Hereafter follows information on the possible status code values.

4.1. Http status codes

```
200 - Measurment data was received successfully.

400 - Invalid request data.

412 - No inspection is currently undergoing.

500 - Internal server error.
```

5. Request/Response examples

REQUEST

Method: POST
Request body:

```
{
    "location": "Sofia Test Str 16",
    "conductedDatetime": "2020-04-23T18:25:43",
    "manufacturer": "Test Company LTD",
    "model": "Test model",
    "serialNumber": "123456",
    "softwareVersion": "1.0",
    "deviceRemark": "remark",
    "regNum": "CA0000AA",
    "measurements": [{
      "rpmResult": 2045,
      "engineOilTemperature": 83,
      "copercent": 0.1,
      "co2Percent": 19.3,
      "o2Percent": 2.4,
      "airRatioLambda": 10.5,
      "hc": 34,
      "fuelType" : 1
   }]
}
```

RESPONSE

Response when the measurment data was received successfully

```
HTTP status: 200
Response body:
{

}
```

Response when there is no currently undergoing inspection

```
HTTP status: 412

Response body:

{

  "error": "There is no currently undergoing inspection"

}
```

Response when the server suffered an internal error

```
HTTP status: 500

Response body:

{

    "error": "Something went wrong"

}
```

II. Предаване на данни от димомера

1. Description

The provided REST service transmits measurement data from an opacity meter to an Executive Agency "Automobile Administration"'s client. The opacity meter sends a request with the measured data and the EA "AA"'s client responds with a http status code.

2. Service endpoints

A request to this endpoint **must contain** the complete set of measured data.

```
Request method: POST

Request URL: /api/inspection/current/measurement/opacity-smoke
```

3. Request

The request consists of the following data:

- **`location`** - Text - The location address where the measurement was conducted.
- **`conductedDatetime`** - Date in ISO 8601 format - The end date time of the measurement.
- **`manufacturer`** - Text - The measuring device's manufacturer name or brand name.
- **`model`** - Text - The measuring device's model.
- **`serialNumber`** - Text - The measuring device's serial number.

- **`softwareVersion`** - Text - The measuring device's software version. (optional)
- **`deviceRemark`** - Text - Additional remarks for the measuring device. (optional)
- **`regNum`** - Text in latin letters and numbers - The vehicle's registration number.
- **`measurements[]`** - An array of performed inspections. The first index of the array corresponds to test 1, the second to test 2 and so on. (length 4)
  An element of the array measurements contains an object with the following structure:
  - **`opacitySmokeResult`** - Number - Light transmission coefficient.
  - **`rpmResult`** - Number - Revolution per minute.
  - **`engineOilTemperature`** - Number - Engine oil temperature.

4. Response

The response of the EA "AA"'s client represents an http status code. Hereafter follows information on the possible status code values.

4.1. Http status codes

```
200 - Measurment data was received successfully.

400 - Invalid request data.

412 - No inspection is currently undergoing.

500 - Internal server error.
```

5. Request/Response examples

REQUEST

Method: POST
Request body:

```
{

    "location": "Sofia Test Str 16",

    "conductedDatetime": "2020-04-23T18:25:43",

    "manufacturer": "Test Company LTD",

    "model": "Test model",

    "serialNumber": "123456",

    "softwareVersion": "1.0",

    "deviceRemark": "remark",

    "regNum": "CA0000AA",

    "measurements": [{
        "opacitySmokeResult": 1.2,
        "rpmResult": 1892,
        "engineOilTemperature": 3.4
    }]
}
```

RESPONSE

Response when the measurment data was received successfully

```
HTTP status: 200
Response body:
{

}
```

Response when there is no currently undergoing inspection

```
HTTP status: 412
Response body:
{
  "error": "There is no currently undergoing inspection"
}
```

Response when the server suffered an internal error

```
HTTP status: 500
Response body:
{
    "error": "Something went wrong"
}
```

III. Предаване на данни от стенда за измерване на спирачните сили на пътните превозни средства

1. Description

The provided REST service transmits measurement data from a brake tester device to an Executive Agency "Automobile Administration"'s client. The brake tester device sends a request with the measured data and the EA "AA"'s client responds with a http status code.

2. Service endpoints

A request to this endpoint **must contain** the complete set of measured data.

```
Request method: POST

Request URL: /api/inspection/current/measurement/brakes
```

3. Request

The request consists of the following data:

- **`location`** - Text - The location address where the measurement was conducted.
- **`conductedDatetime`** - Date in ISO 8601 format - The end date time of the measurement.
- **`manufacturer`** - Text - The measuring device's manufacturer name or brand name.
- **`model`** - Text - The measuring device's model.
- **`serialNumber`** - Text - The measuring device's serial number.
- **`softwareVersion`** - Text - The measuring device's software version. (optional)
- **`deviceRemark`** - Text - Additional remarks for the measuring device. (optional)
- **`regNum`** - Text in latin letters and numbers - The vehicle's registration number.
- **`workingBrakesMeasurements[]`** - Array of objects - Contains results from the tests of each axis's working brake. The first index of the array corresponds to axis 1, the second to axis 2, and so on. (Length 8)
  An element of the array represents an object with the following structure:
  - **`fl`** - Real number - Measured braking force left wheel. *[axis 1]*
  - **`fr`** - Real number - Measured braking force right wheel. *[axis 1]*
  - **`mass`** - Real number - The measured load [N]. *[axis 1]*
  - **`pressure`** - Real number - The measured pressure in the brake pads P [bar]. *[axis 1]*
  - **`rollers`** - Contact Roller - Contact Rollers used in the axis measurement *[axis 1]*

```
o  l - left

o  r - right
```

```
○  b - both
```

- **`handBrakesMeasurements[]`** - Array of objects - Contains results from the tests of each active axis's parking brake. The first index of the array corresponds to axis 1, the second to axis 2, and so on. (Length 8)
  An element of the array represents an object with the following structure:
  - **`fl`** - Real number - Measured braking force left wheel. *[axis 1]*
  - **`fr`** - Real number - Measured braking force right wheel. *[axis 1]*
  - **`mass`** - Real number - The measured load [N]. *[axis 1] (optional)*
  - **`rollers`** - Contact Roller - Contact Rollers used in the axis measurement *[axis 1]*

```
○  l - left

○  r - right

○  b - both
```

4. Response

The response of the EA "AA"'s client represents an http status code. Hereafter follows information on the possible status code values.

4.1. Http status codes

```
200 - Measurment data was received successfully.

400 - Invalid request data.

412 - No inspection is currently undergoing.

500 - Internal server error.
```

5. Request/Response examples

REQUEST

Method: POST
Request body:

```
{

    "location": "Sofia Test Str 16",

    "conductedDatetime": "2020-04-23T18:25:43",

    "manufacturer": "Test Company LTD",

    "model": "Test model",

    "serialNumber": "123456",
    "softwareVersion": "1.0",

    "deviceRemark": "remark",

    "regNum": "CA0000AA",

    "workingBrakesMeasurements": [{

      "pressure": 2.5,

      "fl": 1534.0,

      "fr": 1527.0,

      "mass": 543.0

     " rollers ": "b"

    }],

    "handBrakesMeasurements": [{

      "fl": 1432.0,

      "fr": 1426.0,

      "mass": 543.0

      " rollers ": "b"

    }]

 }
```

RESPONSE

Response when the measurment data was received successfully

```
HTTP status: 200

Response body:

{


}
```

Response when there is no currently undergoing inspection

```
HTTP status: 412

Response body:
```

```
{
  "error": "There is no currently undergoing inspection"
}
```

Response when the server suffered an internal error

```
HTTP status: 500

Response body:

{
    "error": "Something went wrong"
}
```

IV. Предаване на данни в реално време на всяка секунда (опционално, при техническа възможност):

1. Предаване на данни в реално време от газоанализатора (опционално)

1.1. Description

The provided REST service transmits measurement data from an emission gas analyzer to an Executive Agency "Automobile Administration"'s client. The emission gas analyzer sends a request with the measured data and the EA "AA"'s client responds with a http status code.

1.2. Service endpoints

1.2.1. Intermediate live result

A request to the endpoint below **does not need to contain** the complete set of measured data.

```
Request method: POST

Request URL: /api/inspection/current/measurement/gas-emissions/live
```

1.3. Request

The request consists of the following data:

- **`location`** - Text - The location address where the measurement was conducted.
- **`manufacturer`** - Text - The measuring device's manufacturer name or brand name.
- **`model`** - Text - The measuring device's model.
- **`serialNumber`** - Text - The measuring device's serial number.

- **`softwareVersion`** - Text - The measuring device's software version. (optional)
- **`deviceRemark`** - Text - Additional remarks for the measuring device. (optional)
- **`coPercent`** - Real number - Percentage of CO.
- **`co2Percent`** - Real number - Percentage of CO2.
- **`o2Percent`** - Real number - Percentage of O2.
- **`hc`** - Real number - PPM of HC.
- **`rpmResult`** - Number - Revolution per minute.
- **`engineOilTemperature`** - Number - Engine oil temperature.

1.4. Response

The response of the EA "AA"'s client represents an http status code. Hereafter follows information on the possible status code values.

1.4.1. Http status codes

```
200 - Measurment data was received successfully.

400 - Invalid request data.

500 - Internal server error.
```

1.5. Request/Response examples

REQUEST

Method: POST
Request body:

```
{

    "location": "Sofia Test Str 16",

    "manufacturer": "Test Company LTD",

    "model": "Test model",

    "serialNumber": "123456",

    "softwareVersion": "1.0",

    "deviceRemark": "remark",

    "rpmResult": 2045,

    "engineOilTemperature": 83,

    "copercent": 0.1,
    "co2Percent": 19.3,

    "o2Percent": 2.4,

    "hc": 34,

}
```

RESPONSE

Response when the measurment data was received successfully

```
HTTP status: 200

Response body:

{



}

}
```

Response when the server suffered an internal error

```
HTTP status: 500

Response body:

{

    "error": "Something went wrong"

}
```

2 Предаване на данни в реално време от димомера (опционално)

2.1. Description

The provided REST service transmits measurement data from an opacity meter to an Executive Agency "Automobile Administration"'s client. The opacity meter sends a request with the measured data and the EA "AA"'s client responds with a http status code.

2.2. Service endpoints

2.2.1. Intermediate live result

A request to the endpoint below **does not need to contain** the complete set of measured data.

```
Request method: POST
Request URL: /api/inspection/current/measurement/opacity-smoke/live
```

2.3. Request

The request consists of the following data:

- **`location`** - Text - The location address where the measurement was conducted.
- **`manufacturer`** - Text - The measuring device's manufacturer name or brand name.
- **`model`** - Text - The measuring device's model.
- **`serialNumber`** - Text - The measuring device's serial number.
- **`softwareVersion`** - Text - The measuring device's software version. (optional)
- **`deviceRemark`** - Text - Additional remarks for the measuring device. (optional)
- **`opacitySmokeResult`** - Number - Light transmission coefficient.
- **`rpmResult`** - Number - Revolution per minute.
- **`engineOilTemperature`** - Number - Engine oil temperature.

2.4. Response

The response of the EA "AA"'s client represents an http status code. Hereafter follows information on the possible status code values.

2.4.1. Http status codes

```
200 - Measurment data was received successfully.

400 - Invalid request data.

500 - Internal server error.
```

2.5. Request/Response examples

REQUEST

Method: POST
Request body:

```
{
    "location": "Sofia Test Str 16",
    "manufacturer": "Test Company LTD",
    "model": "Test model",
    "serialNumber": "123456",
    "softwareVersion": "1.0",
    "deviceRemark": "remark",
    "opacitySmokeResult": 1.2,
    "rpmResult": 1892,
    "engineOilTemperature": 3.4
}
```

RESPONSE

Response when the measurment data was received successfully

```
HTTP status: 200
Response body:
{

}
```

Response when the server suffered an internal error

```
HTTP status: 500
Response body:
{
    "error": "Something went wrong"
}
```

3. Предаване на данни в реално време от стенда за измерване на спирачните сили на пътните превозни средства (опционално)

3.1. Description

The provided REST service transmits measurement data from a brake tester device to an Executive Agency "Automobile Administration"'s client. The brake tester device sends a request with the measured data and the EA "AA"'s client responds with a http status code.

3.2. Service endpoints

3.2.1. Intermediate live result

A request to the endpoint below **does not need to contain** the complete set of measured data.

```
Request method: POST

Request URL: /api/inspection/current/measurement/brakes/live
```

3.3. Request

The request consists of the following data:

- **`location`** - Text - The location address where the measurement was conducted.
- **`manufacturer`** - Text - The measuring device's manufacturer name or brand name.
- **`model`** - Text - The measuring device's model.
- **`serialNumber`** - Text - The measuring device's serial number.
- **`softwareVersion`** - Text - The measuring device's software version. (optional)
- **`deviceRemark`** - Text - Additional remarks for the measuring device. (optional)
- **`fl`** - Real number - Measured braking force left wheel.
- **`fr`** - Real number - Measured braking force right wheel.

3.4. Response

The response of the EA "AA"'s client represents an http status code. Hereafter follows information on the possible status code values.

3.4.1. Http status codes

```
200 - Measurment data was received successfully.

400 - Invalid request data.

500 - Internal server error.
```

3.5. Request/Response examples

REQUEST

Method: POST
Request body:

```
{
    "location": "Sofia Test Str 16",
    "manufacturer": "Test Company LTD",
    "model": "Test model",
    "serialNumber": "123456",
    "softwareVersion": "1.0",
    "deviceRemark": "remark",
    "fl": 1534.0,
    "fr": 1527.0
 }
```

RESPONSE

Response when the measurment data was received successfully

```
HTTP status: 200
Response body:
{

}
```

Response when the server suffered an internal error

```
HTTP status: 500
Response body:
{
    "error": "Something went wrong"
}
```

V. Получаване на данни

1. Данни за превозното средство

1.1. Description

The following REST service provides information regarding the technical parameters of a vehicle undergoing a technical inspection.

1.2. Service endpoint

```
Request method: GET

Request URL: /api/inspection/current
```

1.3. Request

The request is empty.

1.4. Response

The response consists of an http status code and a response body. Hereafter follows information on the possible status code values and the response body structure.

1.4.1. Http status codes

```
200 - The request was successful.

412 - No inspection is currently undergoing.

500 - Internal server error.
```

1.4.2. Response body

The response **may contain empty/null values**.

- **inspectionId** - Number - Indicates the technical inspection id.

- **vehicle** - Object - Contains information from the registration certificate related to the technical parameters of the vehicle undergoing technical inspection.
    - **vin** - Text - Vehicle identification number.
    - **regNum** - Text in latin letters and numbers - Vehicle registration number.
    - **category** - Text - Vehicle category (nomenclature according The Road Traffic Law).
    - **firstRegistrationDate** - Date in ISO 8601 format – First registration date.
    - **fuelType** - Text - Fuel type.
    - **environmentalCategory** - Text - Environmental category.
    - **measurements** - Object - Contains information about possible measurements/tests.
        - **brakesParameters** - Object - Contains information about the technical parameters of the vehicle related to the braking system check
            - **numberOfAxles** - Number - Number of axles.
            - **brakeSystem** - Object - Contains information about the braking system. The object is empty (null) if no measurement required.
                - **minBrakeSystemEfficiency** - Number - Minimum value for braking system efficiency
                - **pneumaticBrakePressure** - Number - Pressure for pneumatic braking system (bar) defined by the vehicle manufacturer
                - **maxPermitedLadenMass** - Number - Maximum permitted mass kg

  - **systemType** - Number - Indicating the type of the braking system.

    ```
    ▪       1 - A vehicle with mechanical,
       hydraulic and pneumatic brake actuators
       (without pressure regulation in the brake
       pads (cylinders))

    ▪       2 - A vehicle (composition of vehicle
       and trailer, semi-trailer) with pneumatic
       brake system with pressure regulation in the
       brake pads (cylinders)
    ```

- **handBrakeSystem** - Object - Contains information about the hand braking system. The object is empty (null) if no measurement required.
  - **minHandBrakeSystemEfficacy** - Number - Minimum value for hand braking system efficiency
  - **maxBrakingForceInequality** - Number - Maximum permitted inequality of the braking force on the axle for the braking system and the hand braking system
  - **f2 maximum permissible laden mass of the vehicle in service**
  - **f3 maximum permissible laden mass of the whole vehicle in service**
- **opacitySmokeParameters** - Object - Contains information about the technical parameters of the vehicle related to the opacity check
  - **maxOpacityValue** - Number - Maximum permitted values of the light transmission coefficient
  - **minEngineOilTemperature** - Number - Minimum engine oil temperature.
  - **engineType** - Number - Engine type

  ```
  ▪     1 - for naturally aspirated engines
     (without turbocharger):

  ▪     2 - for turbocharged engines (with
     turbocharger):
  ```

- **gasEmissionsParameters** - Object - Contains information about the technical parameters of the vehicle related to the exhaust gases
  - **maxCOValue** - Number - Maximum permitted values of CO - idle
  - **minEngineOilTemperature** - Number - Minimum engine oil temperature
  - **maxCOValue_hight** - Number - Maximum permitted values of CO – min 2000 rpm
  - **minRPM** - Number - Minimum revolution per minute
  - **engineType** - Number - Engine type

  ```
  ▪     1 - Motor vehicles without emission control
     system (without catalytic converters)

  ▪     2 - Motor vehicles with emission control
     system (with catalytic converters)
  ```

1.5. Request/Response examples

REQUEST

Method: GET
Parameters: None

RESPONSE

Response when a valid ungoing technical inspection is found

```
HTTP status: 200

Response body:

{

    "inspectionId": "3245234",

    "vehicle": {

        "vin": "TMBMS46Y864551234",

        "regNum": "CA3456CB",

        "category": "M",

        "firstRegistrationDate": "2011-12-24",

        "fuelType": "Diesel",

        "environmentalCategory": "Euro5",

        "measurements": {

            "brakesParameters": {

                "numberOfAxles": 4,

                "brakeSystem": {

                    "systemType": 1,

                    "minBrakeSystemEfficacy": 1,

                    "pneumaticBrakePressure": 4,

                    "maxPermitedLadenMass": 3000

                },

                "handBrakeSystem": {

                    "minHandBrakeSystemEfficacy": 2,

                    "maxBrakingForceInequality": 0.3
                }
            },
            "opacitySmokeParameters": {
                "engineType": 2,
                "maxOpacityValue": 323,
                "minEngineOilTemperature": 78.3
            },
            "gasEmissionsParameters": {
                "engineType": 1,
                "maxCOValue": 23,
                "minEngineOilTemperature": 78.9,
                "minRPM": 950
            }
        }
    }
}
```

Response when there is no currently undergoing inspection

```
HTTP status: 412
Response body:
{
  "error": "There is no currently undergoing inspection"
}
```

Response when the server suffered an internal error

```
HTTP status: 500
Response body:
{
    "error": "Something went wrong"
}
```

2. Данни от OBD на превозното средство

2.1. Description

The following REST service provides information regarding the current oil temperature and rpm of a vehicle undergoing a technical inspection.

2.2. Service endpoint

```
Request method: GET

Request URL: /api/inspection/current/live
```

1.3. Request

The request is empty.

2.4. Response

The response consists of an http status code and a response body. Hereafter follows information on the possible status code values and the response body structure.

2.4.1. Http status codes

```
200 - The request was successful.

412 - No inspection is currently undergoing.

500 - Internal server error.
```

2.4.2. Response body

The response **may contain empty/null values**.

- `oilTemp` - Real number - The oil temperature.
- `rpm` - Number - The revolutions per minute.

2.5. Request/Response examples

REQUEST

Method: GET
Parameters: None

RESPONSE

Response when a valid ungoing technical inspection is found

```
HTTP status: 200

Response body:

{
    "oilTemp": 86.8,

    "rpm": 1459

}
```

Response when there is no currently undergoing inspection

```
HTTP status: 412

Response body:

{

  "error": "There is no currently undergoing inspection"

}
```

Response when the server suffered an internal error

```
HTTP status: 500

Response body:

{

    "error": "Something went wrong"

}
```

VI. Удостоверяване

All services in test and production environment will require basic authentification.
For the test environment, the following credentials could used:
user.name=techinspDeveloper
user.pass=zVrWzkPAHVBQz83KTeQruBrVHyHxSn
For the production environment each approved software platform will be provided with it’s own credentials.

2. Заповедта да се доведе до знанието на:

2.1. Лицата, които са пуснали или пускат на пазара и/или в действие газоанализатори, димомери и стендове за измерване на спирачните сили на пътните превозни средства, чрез писмо от мое име, подготвено от директора на дирекция „Пътни превозни средства“;

2.2. Директора на дирекция „Пътни превозни средства“, който да запознае със заповедта служителите от дирекцията;

2.3. Директорите на Регионалните дирекции „Автомобилна администрация“, които да запознаят със заповедта служителите, на които е възложено извършването на проверки и контрол на дейността по извършване на периодичните прегледи за проверка на техническата изправност на пътните превозни средства.

3. Заповедта да се публикува на електронната страница на Изпълнителна агенция „Автомобилна администрация“ за сведение на заинтересованите лица.

4. Контрола по изпълнението на заповедта възлагам на заместник изпълнителния директор.

Борислав Муеров
Изпълнителен директор