



# Installatiehandleiding

**Naam:** Yassine Azdad  
**Studentnr.:** 8000153  
**Onderdeel:**  
Installatiehandleiding  
**Vak:** backend  
**Docent:** Nick Stuivenberg

## Inhoudsopgave

Inleiding .....	blz.1.
Installatie .....	blz. 2.
Inloggegevens .....	blz. 3.
Json en rest-endpoint .....	blz. 4 t/m 10.

## **Inleiding**

In deze installatiehandleiding wordt u uitgelegd hoe de applicatie kunt gebruiken. Hierin worden drie verschillende onderdelen behandeld die nodig zijn om de applicatie te kunnen gebruiken. Als eerste wordt u meegenomen in de installatie proces voor de benodigde tools die nodig zijn. In het tweede onderdeel worden u de inloggegevens verstrekt van de gebruikers. In het laatste onderdeel krijgt u de end points toe gerijkt.

# Installatie

Benodigde software:

- Mysql,

Link site:

<https://dev.mysql.com/downloads/>

handige Youtube link:

<https://www.youtube.com/watch?v=18VboHP1YCs>

Note: Als u dat gedaan heeft moet u een schema: garage aanmaken in mysql workbench.

Onthoudt u root wachtwoord, want die heeft uw later nodig bij de property in de applicatie. In de property moet u namelijk het wachtwoord van root naar uw root wachtwoord veranderen. Tevens moet u een query garage aanmaken.

- IDE (Bijvoorbeeld IntelliJ)

De applicatie maakt gebruik van jdk 11 en er is gebruik gemaakt van intelij IDE.

- postman

In postman kunt u de applicatie draaien via :<https://localhost:8443/> en in de selfmade certificaat accepteren. Mocht de gebruiker Postman gebruiker dan bij "File>Settings>SSL certificate verification" op "off" zetten.

- Github (broncode)

Via deze link kunt u eenvoudig naar de applicatie gaan. Tevens staan daar ook de functionele en technische documentatie. [www.github.com/yassine071/autogarage](https://www.github.com/yassine071/autogarage)

## Inloggegevens

Top de benodigde tools zijn gedownload en zijn klaar om gebruikt te worden!

Hiervoor heeft u nog enkele gegevens nodig, want de applicatie is beveiligd en enkel toegankelijk voor de gebruikers met specifieke bemachtiging. Hieronder vindt u de inloggegevens en waartoe de gebruikers gemachtigd zijn.

Admin account:

Username: admin

Password: password

Toegang tot: user, reparation, examination, car, authenticated

User account:

Username: monteur

Password: password

Toegang tot: reparation, examination, component, car, authenticated, action

User account:

Username: back

Password: password

Toegang tot: component, authenticated

User account:

Username: kassa

Password: password

Toegang tot: component, authenticated

## Json en rest-endpoint

Hier worden de json en rest-endpoints gedeeld. Dit kunt uw gebruiken in postman om te testen of de code correct werkt.

### # Endpoints/Json\*\*\*\*\*

// Hiermee haalt u alle users op

\* GET /user

<https://localhost:8443/user>

// hiermee maakt u een nieuwe user aan met de bij behorende gemachtigde rol

\* POST user

<https://localhost:8443/user>

JSON:

```
{
  "username": "user",
  "password": "password",
  "authorities": [
    {
      "username": "user",
      "authority": "ROLE_USER"
    }
  ]
}
```

//hiermee haalt u alleen specifiek de user door username te veranderen naar des betreffende user.

\* GET user/{username}

<https://localhost:8443/user/user>

//hiermee update u de gegevens van de user door username te veranderen in de desbetreffende user. Onderstaand is de wachtwoord gewijzigd van user.

\* PUT user/{username}

<https://localhost:8443/user/user>

JSON:

```
{
  "password": "password1" }
```

//hiermee haalt u specifiek de rol van de user

\* GET /user/{username}/authoritiy

https://localhost:8443/user/user/authority

//door de authority te veranderen naar desbetreffende authority verwijderd u dit.

\* DELETE /user/{username}/authority/{authority}

https://localhost:8443/user/userTest/authorities/ROLE\_USER

\*\*\*\*\*

//hiermee haalt u de action lijst op

\* GET /action

https://localhost:8443/action

// hiermee maakt een nieuwe action aan

\* POST /action

https://localhost:8443/action

JSON:

```
{  
  "price": 500.0,  
  "action": "motor is changed",  
  "name": "motor"  
}
```

//hiermee haalt u alleen specifiek de action door de id te veranderen naar des betreffende id.

\* GET action/{id}

https://localhost:8443/action/5

// door de id te kiezen van de desbetreffende action kunt u dit updaten.

\* PUT /action/{id}

https://localhost:8443/action/5

JSON:

```
{ "price": 450.0,  
  "action": "motor is changed, but needs repair again",  
  "name": "motor" }
```

// door de specifieke id te kiezen die bij de action hoort kunt u dit verwijderen.  
\* DELETE /action/{id}

<https://localhost:8443/action/5>

JSON:

```
{  
  "price": 450.0,  
  "action": "motor is changed, but needs repair again",  
  "name": "motor"  
}
```

\*\*\*\*\*

//hiermee haalt u de cars op.  
\* GET /car

<https://localhost:8443/car>

//hiermee maakt u de car aan.  
\* POST /car

<https://localhost:8443/car>

JSON:

```
{  
  "brand": "volkswagen",  
  "customer_name": "Jan Hendrik",  
  "License_plate": "45-io-89",  
  "model": "polo"  
}
```

// hiermee haalt u de specifieke car op door de id te selecteren die bij de desbetreffende car hoort  
\* GET /car/{id}

<https://localhost:8443/car/3>

JSON:

```
{  "brand": "volkswagen",  
    "customer_name": "Jan Hendrik",  
    "License_plate": "45-io-89",  
    "model": "polo"  }
```



//hiermee update je de gegevens van de desbetreffende car op door de id te selecteren van de car.

\* PUT /car/{id}

<https://localhost:8443/car/3>

```
{  
  "brand": "volkswagen",  
  "customer_name": "Jan Hendrik",  
  "License_plate": "45-io-89",  
  "model": "golf"  
}
```

//door de specifieke id te kiezen kunt u de desbetreffende car verwijderen

\* DELETE /car/{id}

<https://localhost:8443/car/3>

.....

// hiermee haal je alle gegevens van de aanwezige componenten aan.

\* GET /component

<https://localhost:8443/component>

//hiermee voeg je een nieuwe component aan.

\* POST /component

<https://localhost:8443/component>

JSON:

```
{  
  "component": "luchtfILTER",  
  "price": 35.0,  
}
```

//hiermee haalt u de specifieke component aan door de id aan te geven van de desbetreffende component.

\* GET /component/{id}

<https://localhost:8443/component/3>

JSON:

```
{  
  "component": "luchtfiler",  
  "price": 35.0,  
}
```

// hiermee veranderd u de status van de desbetreffende component.

\* PUT /component/{id}

<https://localhost:8443/component/3>

JSON:

```
{  
  "component": "luchtfiler",  
  "price": 45.0,  
}
```

\* DELETE /component/{id}

<https://localhost:8443/component/3>

\*\*\*\*\*

//hiermee haalt u de laast van de examination op.

\* GET /examination

<https://localhost:8443/examination>

//hiermee maakt u een nieuwe examination aan.

\* POST /examination

<https://localhost:8443/examination>

```
{  
  "date": "13-11-2020",  
  "appointment": "audi",  
  "costumer": "user",  
}
```

//hiermee haalt u de desbetreffende examination op.

\* GET /examination/{id}

<https://localhost:8443/examination/3>

```
{  
  "date": "13-11-2020",  
  "appointment": "audi",  
  "costumer": "user",  
}
```

//hiermee wijzigt u de examination.

\* PUT /examination/{id}

<https://localhost:8443/examination/3>

```
{  
  "date": "17-11-2020",  
  "appointment": "audi",  
  "costumer": "user",  
}
```

//hiermee verwijderd u de desbetreffende examination

\* DELETE /examination/{id}

<https://localhost:8443/examination/3>

\*\*\*\*\*

//hiermee haal je de reparation lijst op

\* GET /reparation

<https://localhost:8443/reparation>

//hiermee maakt u een nieuwe reparation aan.

\* POST /reparation

<https://localhost:8443/reparation>

JSON:

```
{  
  "status": " bezig",  
  "date": "17-11-2020",  
}
```

//hiermee haalt u de desbetreffende reparation op.

\* GET /reparation/{id}

<https://localhost:8443/reparation/3>

JSON:

```
{  
  "status": " bezig",  
  "date": "17-11-2020",  
}
```

//hiermee wijzigt u de desbetreffende reparation.

\* PUT /reparation/{id}

<https://localhost:8443/reparation/3>

JSON:

```
{  
  "status": " gedaan",  
  "date": "17-11-2020",  
}
```

//hiermee verwijdert u de desbetreffende reparation.

\* DELETE /reparation/{id}

