

# Installationsguide för Baga Aqua System API

---

2019-05-21

## Förord

---

Denna Installationsguide är för Baga Aqua Systems API. Systemet är byggt i NodeJS med MongoDB som databas. Arbetet är utfört av studenter på Blekinge Tekniska Högskola som en del i kursen PA1416 under vårenterminen 2019 åt Baga Water Technology AB. Denna guide är till för att underlätta vid installation av systemet. Läsaren förväntas ha kunskaper om den egna miljön som installationen sker i.

## Innehållsförteckning

---

- Förord
- Innehållsförteckning
- Microsoft Windows
  - Hämta programvaran
    - Direkt nedladdning
    - Versionshantering med Git
  - Installera databasen
    - Importera data till databasen
    - Konfigurera tillgång till databasen
  - Installera NodeJS
    - Hämta npm beroenden
  - Konfigurera server
    - Anslut till databas
    - Lägg till mailkonto
  - Starta Server
    - Starta i terminal
    - Tillåt servern i brandväggen
  - Underhåll och uppdateringar
- GNU/Linux
  - Hämta programvaran
  - Installera NodeJS
  - Installera MongoDB
  - Hämta npm beroenden
  - Kör konfigurationsskripten
  - Underhåll och uppdateringar

# Installera för Microsoft Windows

## Hämta programvaran

Det första steget i installationen är att hämta programvaran från [GitHub](#). Detta kan göras på två sätt - direkt nedladdning eller via Git.

### Direkt nedladdning

Gå till <https://github.com/vattenbagarna/Backend> och klicka på knappen `clone or download`, klicka sedan på `download zip`. Ladda ner och spara zip-filen på servern. Extrahera zip-filen till en mapp. Du ska nu ha en mapp med alla filerna som servern behöver. Kom ihåg var den är vi återkommer till den senare under installationen.

### Versionshantering med Git

Om du har installerat git på din server så går det aningen lättare. Då öppnar du din git-terminal, navigerar till mappen där du vill ha filerna och klonar repot med följande kommando: `git clone`

`https://github.com/vattenbagarna/Backend.git`

Du ska nu ha en mapp med alla filerna som servern behöver. Kom ihåg var den är vi återkommer till den senare under installationen.

## Installera databasen

Databasen som används är [MongoDB](#) och kan hämtas från deras Download-sida:

<https://mongodb.com/download-center/community>. Hitta rätt version för ditt operativsystem och ladda ner installationsprogrammet. Kör installationsprogrammet och följ instruktionerna på skärmen. Se till att välja alternativen installera som service och Kör service som "network service user". Dessa alternativen brukar vara ikryssade från början.

### Importera data till databasen

I mappen som vi hämtade tidigare så finns lite data att utgå ifrån, den måste nu importeras i databasen.

Starta programmet MongoDB från mappen `C:\Program Files\MongoDB\Server\4.0\bin\mongo`.

Programmet kommer att starta en MongoDB terminal.

Börja med att skapa databasen: `use бага`. Detta kommer skapa en databas som heter бага och den kommer att användas i fler kommandon längre ner. Om du vill döpa din databas till något annat, byt ut бага här och i resten av dokumentet mot ditt nya databasnamn.

Därefter ska det läsas in data i databasen. Använd filsökvägen till din nedladdade mapp som `<din_mapp>` och kör följande kommandon:

```
load("<din_mapp>/Backend-master/setup/createCollections.js")
```

```
load("<din_mapp>/Backend-master/setup/insertObjects.js")
```

*Notera att exakta filsökvägen kan se annorlunda om du har klonat repot via git.*

## Lägg till databasaccess för server

För att att NodeJS servern ska ha tillgång att läsa från databasen behövs databaskonton. För att skapa ett sådant konto, skriv följande kommandon i MongoDB-terminalen.

```
use бага
```

```
db.createUser({user: <systemetsAnvändarnamn>, pwd: <systemetsLösenord>, roles: [{role: readWrite, db: бага}]})
```

Byt ut `<systemetsAnvändarnamn>` mot användarnamnet du vill att systemet ska ha. Byt ut `<systemetsLösenord>` mot lösenordet som systemet ska använda. **OBS!** Tänk på att spara detta lösenord då det kommer behövas senare.

Gör nu samma sak igen men byt ut `бага` mot `admin` för att skapa ett administrativt konto till databasen. Detta konto kan användas om en administratör till exempel behöver redigera databasen manuellt.

## Installera NodeJS

Gå till <https://nodejs.org/en/download/> och hämta installationsprogrammet för NodeJS LTS för ditt operativsystem. Kör installationsprogrammet och följ instruktionerna på skärmen.

## Konfiguration av systemet

Många av systemets inställningar görs i filer, dessa måste konfigureras manuellt när man kör Windows.

### dbConfig.js

Börja med att byta namn på `dbConfig.example.js` till `dbConfig.js` i mappen `config` och öppna den för redigering. Detta kan ske många olika program så som Atom, VSCode, Notepad++ eller om inget annat finns att tillgå, notepad.exe. I filen finns 4 fält som ska redigeras. "base\_url" är var databasen kan nås. I de flesta fallen så kör databasen på samma maskin som NodeJS systemet. När så är fallet så ska den ha värdet "localhost".

Fältet "Database" har i exemplet "YourDatabase". Detta ska bytas ut mot vad din databas heter. I den här instruktionen har vi skapat databasen "бага".

Slutligen har vi fälten "username" och "password" här ska du fylla i användarnamn och lösenord som du skapade i stycket "[Lägg till databasaccess för server](#)".

### emailConfig.js

Byt namn på filen `dbConfig.example.js` till `dbConfig.js` i mappen `config` och öppna den för redigering. Här ska du skriva in address, port, mailaddress och lösenord för kontot som systemet skickar mail ifrån. Vilka värden du ska skriva i beror helt och hållet på vad för mailsystem du själv använder.

## Initiera systemet

Starta windows kommandotolk (nedan cmd) och navigera till din mapp. Du kan även starta cmd direkt i mappen via fältet för filsökvägen eller shift+högerklick. Efter det skriver du följande kommando: `npm install`. När det är klart kör du kommandot `cd setup` följt utav kommandot `node createAdminAccount.js`. Detta kör ett skript som hjälper dig att skapa en första användare. Skriptet är bara tänkt att köras en gång och den som har det administrativa kontot kan sedan bjuda in både administratörer och vanliga användare.

## Starta Server

---

### Starta i terminalen

För att starta servern ska du ha öppnat cmd i serverkatalogen och kör kommandot `node index.js`. Detta startar servern i denna terminalen. För att avsluta / stoppa servern trycker du på tangentbordet `Ctrl + c`. Notera att den inte kommer starta automatiskt om ni stoppar den eller startar om värdmaskinen.

### Tillåt servern i brandväggen

När man startar servern för första gången så brukar Windows be en ge systemet tillgång till internet, godkänn detta. I övrigt kan du behöva justera dina brandväggar och öppna portar beroende på hur ditt nätverk ser ut.

## Underhåll och uppdateringar

---

Hela projektet finns på GitHub med historik av alla förändringar. Det är rekommenderat att bibehålla git versionshantering för projektet ifall det i framtiden kommer att vidareutvecklas.

## GNU/Linux

---

### Hämta programvaran

---

```
git clone git@github.com:vattenbagarna/Backend.git
eller
git clone https://github.com/vattenbagarna/Backend.git
```

### Installera NodeJS

---

ubuntu:

```
sudo apt install nodejs npm
```

arch:

```
sudo pacman -S nodejs npm
```

Alternativt hämta binärpaketet från <https://nodejs.org/en/download/>

## Installera MongoDB

---

Installation varierar beroende på distribution. Arch AUR har ett färdigkompilerat paket vid namn mongodbin. För kompletta instruktioner för din distribution se

<https://docs.mongodb.com/manual/administration/install-on-linux/>

## Hämta npm beroenden

---

Stå i mappen `Backend` och kör kommandot `npm install`

## Kör konfigurationsskripten

---

Flytta till mappen `setup`, kör skriptet `setup.sh` och följ instruktionerna på skärmen.

När det är klart så kör du `node createAdminAccount.js` för att skapa den första administratörsanvändaren.

## Underhåll och uppdateringar

---

Hela projektet finns på GitHub med historik av alla förändringar. Det är rekommenderat att bi behålla git versionshantering för projektet ifall det i framtiden kommer att vidareutvecklas.