Samling med tema og videoer for webutvikling

Her finner du en samling med videoer (som ligger på YouTube), som viser ulike ting du kan få behov for på din egen server.

All koden som blir vist ligger i tillegg på Github: <https://github.com/hausnes/1IMA-2024-2025/tree/main/databaser/bildatabase>

Bruk den koden du har behov for – ikke bare kopier alt. Tanken er at du ikke aktiverer ruter og innstillinger som du ikke har behov for, fordi du tenker på sikkerhet.

Innhold

[Node JS og Express 1](#_Toc832293465)

[Sette opp server 2](#_Toc421302282)

[Ruter på egen server 2](#_Toc2063407993)

[Avanserte ruter på egen server 2](#_Toc427975780)

[Hvordan bruke public-mappen 2](#_Toc1103644836)

[Hvordan legge til data i DB via nettside 2](#_Toc1195346323)

[Krypterte data, del 1: Legge til krypterte data i DB 2](#_Toc1836961596)

[Krypterte data, del 2: Logge inn vha. kryptert passord, skjule innhold 3](#_Toc113569355)

[Logge inn, vise meny og dashboard for gitt bruker 3](#_Toc1959252690)

[Slette data som du har lov til (dine egne) 3](#_Toc520744928)

[REST API 3](#_Toc882885739)

[Eksempel: randomuser.me 4](#_Toc1177892202)

[Eksempel: localhost 4](#_Toc21573856)

[Sikkerhet 4](#_Toc1818293847)

[DDoS-eksempel, del 1: Oppsett og angrep 5](#_Toc260247832)

[DDoS-eksempel, del 2: Beskyttelse vha. rate limiting 5](#_Toc1805772975)

# Node JS og Express

## Sette opp server

## Ruter på egen server

Video: <https://youtu.be/3qXJcVHgwB0> (5 min.)

Denne videoen viser hvordan du setter opp en egen Express-server, og deretter åpner ulike ruter. Disse rutene går til ulike sider som henter data fra databasen vi har satt opp.

## Avanserte ruter på egen server

Video: <https://youtu.be/tS0khfvhBqs> (10 min.)

Denne videoen viser hvordan du setter opp og planlegger mer avanserte ruter til serveren din.

## Hvordan bruke public-mappen

Video: <https://youtu.be/y2S_wVNT32g> (2 min.)

Denne videoen viser hvordan du kan lage en public-mappe, og aktivere støtte for dette i app.js-filen din, slik at du ikke lenger bare trenger å peke de besøkende til en rute, men heller til en X.html (med tilhørende bilder, CSS og JS).

## Hvordan legge til data i DB via nettside

Video: <https://youtu.be/Fqi6RsVPOa4> (11 min.)

Denne videoen viser hvordan du kan lage en rute i app.js-filen din som tillater å legge til data i databasen. I tillegg blir det vist hvordan du lager en nettside (i public-mappen), som bruker et skjema (form), litt enkel JS (await, async), og ruten du lagde på serveren, for å kunne sette inn faktiske data.

Merk at det blir vist litt om feilmeldinger og problemer som kan oppstå, men her er det mer arbeid å gjøre.

Merk at HTML, CSS og JS er i en og samme fil i disse eksemplene, for at det skal være lettere å få oversikt i videoen. Disse bør som alltid separeres i egne filer.

## Krypterte data, del 1: Legge til krypterte data i DB

* Video: <https://youtu.be/p8lsmmZfxtM> (8 min.)
* Video om bcrypt (engelsk): ”[Bcrypt & Password Security - An Introduction](https://youtu.be/O6cmuiTBZVs?si=ApK2fZ9GAYVZhZ7Q)”
* Video om bcrypt i Node JS og Express: ” [Hashing Passwords in Node and Express using bcrypt](https://www.youtube.com/watch?v=AzA_LTDoFqY&t=587s)”

Den øverste videoen viser hvordan du kan la brukere registrere seg på nettsiden, med et kryptert passord. Vi bruker bcrypt for å få dette til.

I tillegg viser videoen hvordan du kan unngå feil med fremmednøkler, slik vi så i forrige video. Dette blir håndtert med en ny rute som henter alle postnummer fra databasen, for så at disse blir fylt ut i HTML-en i form av et select-element.

Merk at HTML, CSS og JS er i en og samme fil i disse eksemplene, for at det skal være lettere å få oversikt i videoen. Disse bør som alltid separeres i egne filer.

## Krypterte data, del 2: Logge inn vha. kryptert passord, skjule innhold

Video: [https://youtu.be/pk9SNpc8jR0 (10](https://youtu.be/pk9SNpc8jR0%20(10) min.)

Denne videoen viser hvordan du legger til middleware på serveren som sjekker om du er innlogget før du får besøkt visse ruter og sider. Vær oppmerksom på at dette for HTML-dokumenter krever at du flytter disse fra public-mappen til en annen, og bruker sendFile.

Dette er et viktig eksempel som viser noen av prinsippene.

Merk at HTML, CSS og JS er i en og samme fil i disse eksemplene, for at det skal være lettere å få oversikt i videoen. Disse bør som alltid separeres i egne filer.

## Logge inn, vise meny og dashboard for gitt bruker

Video: <https://youtu.be/Qbqd2jFspVI> (7 min.) (beklager lyden!)

Nye filer:

* app08.js
* dashboard.html (beskytta)
* minebiler.html (beskytta)
* (liten endring i login.html)

I denne videoen utvider vi litt konseptet fra tidligere, med en litt mer logisk måte å bruke tjenesten på. Vi lagar en form for dashboard, der brukeren, etter innlogging, får se valg som er relevante (vis mine biler, legg til bil og logg ut).

## Slette data som du har lov til (dine egne)

Video: ..

Kommer.

# REST API

## Eksempel: randomuser.me

Video: <https://youtu.be/6uBnAGVg7wY?si=eDOwvFHuYVLWXmPT> (14 min.)

Denne videoen viser hvordan du kan besøke åpne API-er, og lage nettsider basert på dette. Et fokus i videoen er hvordan du først får data i **JSON-format**, for så at du forsøker å finne ut hvordan du henter ut det du er interessert i.

Videre så vises det teknikker for å dynamisk legge til nye elementer i DOM-en (HTML), ved å bruke **createElement og appendChild**. Dette er viktige teknikker!

## Eksempel: localhost

Video: <https://youtu.be/kqha32VOwAY> (14 min.)

Denne videoen viser hvordan du aktiverer **CORS** på serveren, slik at hvem som helst kan koble seg til din egen, åpne API, for så å hente ut interessante data.

I tillegg viser videoen hvordan du kan jobbe med å tolke hva du får fra serveren (data i form av JSON), for at du deretter viser frem dette innholdet på HTML-siden.

# Sikkerhet

## DDoS-eksempel, del 1: Oppsett og angrep

Video: <https://youtu.be/GpBxC5wqJxQ> (12 min.)

Denne videoen viser hvordan du setter opp en egen server vha. Node JS og Express. Det vises en teknikk for at du logger de besøkende, slik at vi ser besøkene/«angrepene».

Videre så vises et enkelt Python-script som lar deg kjøre en mengde forespørsler til serveren.

NB: Merk at det er ulovlig å angripe andre maskiner/servere ifølge norsk lov. INGEN TRAFIKK SKAL GÅ UT FRA DIN MASKIN TIL EN ANNEN, LA BÅDE SERVER OG «ANGREP» KJØRE PÅ EGEN MASKIN.

## DDoS-eksempel, del 2: Beskyttelse vha. rate limiting

Video: <https://youtu.be/GMoIkJ2Bfk0> (3 min.)

Denne videoen viser en ENKEL teknikk for å kunne begrense antall forespørsler fra serveren din, slik at du kan sette en grense for hver besøkende.