# Opplæringsmateriell for IT-lærling

# Introduksjon

Dette veiledningsverktøyet vil rette frem mot hva en IT-lærling bør kunne for å videreutvikle på prosjektet mitt.

# Hva bør du ha gjort på forhånd?

|  |  |
| --- | --- |
| **Hva** | **Hvor** |
| Må ha installert et koderedigeringsprogram: Visual Studio Code (anbefalt). | Kan lastes gjennom følgende lenke:  <https://code.visualstudio.com/download/> |
| Installert lamp-stack inkludert phpmyadmin med GUI. | Kan lastes gjennom følgende lenker:    <https://www.digitalocean.com/community/tutorials/initial-server-setup-with-debian-11>  <https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mariadb-php-lamp-stack-on-debian-11>  <https://www.digitalocean.com/community/tutorials/how-to-install-phpmyadmin-from-source-debian-10> |
| Ha satt opp to maskinvarer. En for klient og en annen for server. Operativsystem: Debian 11. |  |

# Hvordan få tilgang til prosjektet mitt?

Gå til følgende lenke: [https://github.com/Sanchay13](https://github.com/Sanchay13/Aarsoppgave). Dette er GitHub profilen min.

Da vil du bli sendt til følgende side: A screenshot of a computer

Description automatically generated

Her kan du se alle filer som jeg har tilgjengelig på prosjektet mitt, dette er en public respository. Deretter velger du alternativet «**code**» og sekundæralternativet «**Download ZIP**»

Et bilde som inneholder tekst

Automatisk generert beskrivelse

Nå vil en ZIP-fil av prosjektet lastes.

For at en skal få tilgang til prosjektet på nettleseren må den ligge på **htdocs/html** mappen. Dette er fordi htdocs er funksjonaliteten som apache bruker for at filer skal servere som standard fra din domene. For å flytte filen så må du gjøre følgende i terminalen:

**Skriv:** sudo mv prosjektfil /var/www/html/

# Hvordan bruke LAMP:

Lamp står for Linux, Apache, MySQL/MariaDB, PHP/Perl/Python. Dette er en åpen kildekode operativsystem som brukes som plattform for webservere. Derfor skal dette installeres kun på server maskinen din. I Linux er det som oftest enklest å jobbe terminalbasert. Derfor er det kjekt å kunne en del Linux kommandoer som kan være nyttig å kunne. Vedlagt ligger en visualisering av dette:

|  |  |
| --- | --- |
| **Linux kommando** | **Beskrivelse** |
| **su "brukernavn"** | For å logge inn som server user |
| **sudo service programnavn start** | For å starte programvaren |
| **sudo systemctl start programnavn** |  |
| **sudo systemctl stop programnavn** |  |
| **Apache** |  |
| **sudo apt update** | oppdaterer apt som brukes for å laste ned elementer |
| **sudo apt install apache2** |  |
| **sudo ufw app list** |  |
| **sudo ufw allow 'WWW'** |  |
| **sudo systemctl status apache2** | statuscheck |
| **sudo systemctl is-enabled apache2** | sjekker om apache er på |
| **hostname -I** | ser hostname som bare er ip-adressen |
| **sudo systemctl stop apache2** |  |
| **sudo systemctl start apache2** |  |
| **sudo systemctl restart apache2** |  |
| **sudo systemctl enable/disable apache2** |  |
| **sudo mv prosjektfil /var/www/html/** | flytter prosjektet fra downloads til htdocs mappa |

# Hvordan starte server:

For å starte serveren så må du gjøre følgende i terminalen skriv følgende. **sudo systemctl start apache2.**

For å sjekke statusen på apache serveren din skriv følgende: **sudo systemctl status apache2**

For å stoppe serveren, gjør følgende: **sudo systemctl stop apache2**

# Hvordan starte prosjektet mitt

For å koble til databasen. Start Apache og MySQL. Trykk dermed på new og lag en ny database ved navn kantine. Etter hvert, trykk på **phpMyAdmin** logoen og dermed trykk på **User** **account**. Trykk på **Add user account.** Fyll dermed ut slik. Husk og huk av at du skal ha alle globale privilegier siden du er en administrator. Husk at neste gang du logger inn på phpmyadmin så må du logge inn med samme brukernavn og passord, som du har laget nå.

A screenshot of a computer

Description automatically generated with medium confidence

# Hvordan få databaseinnholdet mitt til din maskin:

Det er en forskjell å ha backup av koden på databasen og selve innholdet og strukturen som jeg har laget her. Derfor har phpmyadmin laget en funksjonalitet som gjør dette for deg.

På phpMyAdmin siden så skal du trykke på **import**, dermed vil du bli sendt til følgende side:

Et bilde som inneholder tekst

Automatisk generert beskrivelse

På denne siden så velger du filen du har lyst til å importe ved å trykke på **choose file**. I dette tilfellet er filen **kantine.sql**:

A screenshot of a computer

Description automatically generated with low confidence

Denne filen finner du inne på GitHub profilen min. Deretter må du passe på at formatet på filen er **SQL**:

Et bilde som inneholder tekst

Automatisk generert beskrivelse

Når du er fornøyd med alt så trykker på **import**.

Når du skal lage backups av databasen kan du også bruke en sql-backup program som automatisk tar backups av din database og innholdet som er lagret her. Kan lastes gjennom følgende lenke: <https://sqlbackupandftp.com/>.

Dette er en applikasjon som lar oss ta back-up kopier av våre SQL-databaser og lagre dem på for eksempel OneDrive. Denne applikasjonen fungerer ved å koble til SQL serveren og dermed vil den automatisk ta kopier av din database. Uten å gjøre det like tungvint med metoden jeg brukte øverst. Man kan angi tidspunkter på når vi vil at backup-ene skal skje. I tillegg til dette har applikasjonen en funksjonalitet som gjør at du kan kryptere back-upene med passordbeskyttelse slik at den er trygg og sikker under overføring og lagring av data.

# Hvordan få server PC og klient til å kommunisere?

Du trenger to nettverkskabler og en switch som skal kobles på server og klient pc-en.

1. Først så må du sette statisk IP på server pc-en. Slik som dette: Åpne terminalvinduet med snarveien, Ctrl+Alt+T
2. Sjekk nåværende nettverks Interface med følgende kommando: **ip addr**
3. Se hvilken interface du har lyst til å endre og noter ned for senere bruk
4. Disable nåværende nettverk user interface med å kjøre følgende kommando: **sudo ip link set <interface-name> down**
5. konfigurer den nye ip addressen med følgende kommando: **sudo ip addr add <new-ip-address>/<subnet-mask> dev <interface-name>**
6. Bytt ut de områdene merket med <> med det aktuelle navnet du har lyst til å gi dem.
7. Deretter må du enable netwok interface med den nye ip adressen, med følgende kommando: **sudo ip link set <interface-name> up**
8. sjekk om alt har blitt endret med å kjøre følgende kommando: **ip addr**
9. Nå bør alle endringene være lagret.

Deretter så skal du pinge pc-ene for å se om de i det hele tatt kommuniserer med hverandre. Dette gjør du med følgende kommando på ledetekst. Eks: **ping XXX.XX.XXX.X**

# Connection kode

På Connection delen av koden må du huske å endre IP-adressen til server-pc-en til **localhost**, men hvis folk skal gå til din nettside må de taste inn IP-adressen til maskinen din:

A screen shot of a computer

Description automatically generated with medium confidence

Nå bør alt være oppe å kjøre og nettsiden bør være tilgjengelig for bruk.

# Har du ikke tilgang til nettsiden etter alle gjennomførte steg?

**Sjekk at:**

Wifi er skrudd av (du skal ikke ha tilgang til nett).

Sjekk om brannmuren er skrudd av.

**Funker det fortsatt ikke?**

Restart maskinvaren

# Hva tenker jeg å gjøre videre?

Til videre arbeid så tenker jeg å lage en historikk side, der bruker-administratoren skal kunne se hvilke endringer en enkelt bruker har gjort på nettsiden.

Jeg skal også lage en nettprat-funksjonalitet mellom ansatte og bruker administrator siden, der det skal være mulig å sende melding om glemt brukernavn og passord. Dette vil forsterke brukerstøtten på verket mitt.