|  |  |
| --- | --- |
| Prosjektrapport  Oliver Lindtein |  |

Innholdsfortegnelse

[Prosjektplan 2](#_Toc166756940)

[Hvordan fungerer Flask? 2](#_Toc166756941)

[Dokumentasjon 3](#_Toc166756942)

[Backup-rutiner 3](#_Toc166756943)

[Teknisk dokumentasjon 4](#_Toc166756944)

[Ip-adresser + hostname 4](#_Toc166756945)

[Porter 4](#_Toc166756946)

[Server-funksjon 5](#_Toc166756947)

[Brukerveiledninger 5](#_Toc166756948)

[sette opp host: 5](#_Toc166756949)

[sette opp database: 6](#_Toc166756950)

[Relevante lenker 7](#_Toc166756951)

[Github 7](#_Toc166756952)

[Opplæringsmateriale 7](#_Toc166756953)

[Kartlegging av relevant lovverk 7](#_Toc166756954)

[Risikoanalyse og tiltaksplan 7](#_Toc166756955)

[Egenevaluering 8](#_Toc166756956)

# Prosjektplan

Dette prosjektet er en enkel webapplikasjon utviklet med Flask, som lar brukerne søke etter nettsider basert på søkeord. Applikasjonen kobler seg til en MySQL-database som inneholder oppføringer for forskjellige nettsidene. Brukerne kan søke gjennom disse oppføringene basert på tittel, beskrivelse eller innhold.

## Hvordan fungerer Flask?

Flask er et rammeverk som gjør det enkelt å lage nettsider og webapplikasjoner med Python. Flask-routing lar deg bestemme hvilken kode som skal kjøres når en bestemt URL blir besøkt. Du kan også sende og motta data mellom nettsiden din og brukerne ved hjelp av Flask.

# Dokumentasjon

Nettverkstegning

A computer screen shot

Description automatically generated with medium confidence

## Backup-rutiner

Prosjektet mitt ligger både lokalt og på GitHub (public repo). Det skal veldig mye til for at prosjektet forsvinner fra begge stedene. Kanskje en veldig stor meteor eller EMP fra sola?

## 

## Teknisk dokumentasjon

Bruker/Passord

|  |  |  |
| --- | --- | --- |
|  | **Ubuntu** | **Database (mariadb)** |
| **Brukernavn** | oliver | admin |
| **Passord** | password | password |
|  |  |  |

### Ip-adresser + hostname

10.2.2.228/5000

127.0.0.1

Hostname: oliver

### Porter

80

443

3306

5000

### Server-funksjon

Python på serveren tar bruker input og finner innhold fra databasen for å vise fram

## 

## Brukerveiledninger

### sette opp host:

installer nyeste versjon av ubuntu

Du kan bruke denne brukerveiledningen <https://itsfoss.com/install-ubuntu/>

opne terminal med CTRL + ALT + T

sudo apt update (laste ned oppdateringer)

sudo apt upgrade (installer oppdateringer)

sudo apt install python3-Flask ()

flask –version (sjekke om flask er installert)

sudo apt install python3-mysql.connector ()

sudo apt install git (du skal bruke dette for å laste ned prosjektet mitt)

sudo git clone <url til github repo> (dette vil klone prosjektet til home directory)

hvis man kjører python filen så skal nettsiden hostes (python3 main.py)

**hvis du får «access denied» for db brukeren kan du gjøre følgene**

sudo mysql -u <brukernavn> -p

grant all privileges on web\_pages\_db.\* to “<brukernavn>”@”<host>”

### 

### sette opp database:

laste ned mariadb/sql-server

følg denne brukerveiledningen: (<https://www.digitalocean.com/community/tutorials/how-to-install-mariadb-on-ubuntu-20-04>)

lage bruker som heter «admin» og har passordet «password»

velg eget brukernavn og passord hvis du ikke tester løsningen

lim inn dbcreate.sql scriptet fra repoet for å opprette database og tabeller

lim inn testdata.sql scriptet som ligger i repoet for å legge inn testdata

endre database login credentials I main.py hvis nødvendig

## 

## Relevante lenker

### Github

<https://github.com/olilinvar/searchengine>

### Opplæringsmateriale

(Video kommer straks!!)

# Kartlegging av relevant lovverk

IP adressene som besøker vil dukke opp på hosten. IP adresse kan i visse tilfeller anses som personlig opplysning ifølge GDPR-loven. Åndsverksloven gjelder også for programvare. Universell utforming er viktig for at alle kan bruke siden.

# Risikoanalyse og tiltaksplan

Løsningen min håndterer ikke sensitiv data slik den er i dag. Derfor er det veldig lav eller ingen risiko.

Mitt forsøk på å reflektere over risiko:

Løsningen min er ikke DDOS beskyttet. (Veldig usannsynlig. Konsekvens: Siden funker ikke)

Noen kan trenge seg inn på maskinen og endre nettsiden slik at dem som besøker klikker på farlige lenker. (Veldig usannsynlig. Konsekvens: setter andre for dataangrep. Da er jeg ansvarlig)

Noen trenger seg inn i databasen. (Veldig usannsynlig. Konsekvens: De får se alle oppskriftene)

SQL Injection (Veldig usannsynlig. Jeg kan ha «prepared statements» for å beskytte mot dette)

HTTP. Ikke HTTPS.

# Egenevaluering

## Utfordringer?

Jeg hadde egentlig en samarbeidsoppgave med Trygve som vi jobbet på. Vi bestemte oss for å splitte slik at vi hadde hver vår oppgave. Da måtte jeg begynne på nytt og det ble derfor litt dårlig tid (2/3 uker). Jeg er ansvarlig for dette og skulle ha løsrevet meg tidligere eller startet med et solo prosjekt hjemme.

## Lært noe?

Ja! Har aldri brukt flask før nå. Det er en enkel måte å sette opp en webserver og jeg kan tenke meg å bruke det mer i framtiden.

## Vurder eget arbeid

Kunne gjort ting annerledes. Eller har jeg vært effektiv i perioden jeg har jobbet med prosjektet. Jeg er motivert til å fortsette med lignende prosjekter hjemme.

## Hva burde ha blitt gjort annerledes?

Nevnte noen ting i utfordringer-delen. Jeg burde ha kommet i gang med mitt nåværende prosjekt tidligere. Jeg kunne ha tatt et Python Flask kurs for å bygge en bedre forståelse for flask. Det var dumt av meg å komme i gang sent med brukerstøtte. Jeg burde ha videreutviklet løsningen min. Den er litt for enkel. Jeg kan vise litt av mitt forrige prosjekt med Trygve hvis det hjelper. Vi hadde veldig god dokumentasjon. Igjen, jeg kunne ikke noe flask for to uker siden og startet fra bunnen av på nytt.

# Kilder

W3 schools

Stack Overflow

Alf-Morten, Nora og Erik

YouTube

ChatGPT

Bing AI