Cosmin-Ionel Enachescu

Imi shitpoststatus

Vår 2024

Table of Contents

[Prosjektrapport for IMI Shitpoststatus 2](#_Toc166580154)

[Prosjektbeskrivelse 2](#_Toc166580155)

[**Motivasjon** 2](#_Toc166580156)

[Kort om teknologivalg 2](#_Toc166580157)

[Eventuelt samarbeid 2](#_Toc166580158)

[Eksterne avhengigheter 2](#_Toc166580159)

[Prosjektplan 2](#_Toc166580160)

[Dokumentasjon 3](#_Toc166580161)

[**Nettverkstegning** 3](#_Toc166580162)

[Backup-rutiner 3](#_Toc166580164)

[Teknisk dokumentasjon 3](#_Toc166580165)

[Brukerveiledninger 4](#_Toc166580166)

[Relevante lenker 4](#_Toc166580167)

[Kartlegging av lovverk 4](#_Toc166580168)

[**Hvilke lover og regler må dere som systemeiere kjenne til?** 4](#_Toc166580169)

[**Hvilke grep må gjøres for å oppfylle lovverket?** 4](#_Toc166580170)

[Risikoanalyse og tiltaksplan 4](#_Toc166580171)

[**Risikoanalyse** 4](#_Toc166580172)

[**Tiltaksplan** 4](#_Toc166580173)

[Egenevaluering 5](#_Toc166580174)

[**Har du vært effektiv** 5](#_Toc166580175)

[**Hva har du slitt med** 5](#_Toc166580176)

[**Hvor har du gjort det bra** 5](#_Toc166580177)

[**Hva har du blitt bedre på** 5](#_Toc166580178)

[**Hva ville du gjort annerledes** 5](#_Toc166580179)

[Veiledning for lærlinger 5](#_Toc166580180)

[Mer om lovverk 7](#_Toc166580181)

[Database 8](#_Toc166580182)

[Database Schema for IMISHITPOSTSTATUS 8](#_Toc166580183)

# Prosjektrapport for IMI Shitpoststatus

## Prosjektbeskrivelse

**Motivasjon**

Formålet med IMI Shitpoststatus (tidligere kjent som cars4u) er å skape en plattform hvor alle kan diskutere og dele informasjon om shitpost. Nettstedet vil tilby en brukerengasjerende opplevelse med funksjoner som innlegg, likes, og kommentarer.

## Kort om teknologivalg

Jeg har valgt følgende teknologier for å bygge nettstedet:

* **Backend:** PHP 8.2
* **Frontend:** HTML, CSS, JavaScript, AJAX
* **Database:** MySQL
* **Server:** Debain, Apache
* **Virtualisering:** Hyper-V Manager
* **Fjernstyring:** SSH via Git BASH

## Eventuelt samarbeid

Prosjektet er utført individuelt uten samarbeidspartnere.

## Eksterne avhengigheter

* MySQL Database Server
* Apache Web Server
* PHP 8.2
* Hyper-V Manager for virtualisering
* SSH via Git BASH for fjernstyring

## Prosjektplan

* **Januar 2024:** Oppsett av utviklingsmiljø og grunnleggende serverkonfigurasjon.
* **Februar 2024:** Implementering av brukerregistrering og innloggingssystem.
* **Mars 2024:** Opprettelse av innleggssystem, med mulighet for å legge til bilder og beskrivelser.
* **April 2024:** Implementering av likes/dislikes funksjonalitet og forberedelse til live presentasjon.
* **Mai 2024:** Utvikling av svarsystem for innlegg og forbedring av eksisterende funksjoner + testing.

# Dokumentasjon

**Nettverkstegning**

**A computer screen shot of a computer

Description automatically generated**

## Backup-rutiner

* **Månedlige backups** av MySQL databasen til en lokal disk.
* **En backups** av hele serveren til en ekstern harddisk.
* **GitHub**

## Teknisk dokumentasjon

* **Server:** DEBIAN-serveren er vertikalisert ved bruk av Hyper-V Manager og kjører Apache for webserveren.
* **Database:** MySQL-MariaDB er brukt som databasesystem. Strukturen består av tabellene posts og users.
* **Backup:** Beskrevet i backup-rutinene.

## Brukerveiledninger

* **Oppsett av utviklingsmiljø:** Veiledning for installasjon av nødvendige verktøy og oppsett av server.
* **Brukerregistrering:** Trinnvis prosess for å registrere en ny bruker og logge inn.
* **Oppretting av innlegg:** Instruksjoner for å opprette og administrere innlegg.
* **Alle ligger i en mappe på GitHub**

## Relevante lenker

* Dokumentasjon for PHP 8.2: [php.net](https://www.php.net/releases/8.2/en.php)
* MySQL Referansemanual: [dev.mysql.com](https://dev.mysql.com/doc/)
* Apache HTTP Server Documentation: [httpd.apache.org](https://httpd.apache.org/docs/)

# Kartlegging av lovverk

**Hvilke lover og regler må dere som systemeiere kjenne til?**

* **GDPR (General Data Protection Regulation):** Beskyttelse av personopplysninger.
* **Ekomloven:** Regler om elektronisk kommunikasjon.
* **Opphavsrett:** Beskyttelse av intellektuell eiendom og innhold på nettstedet.

**Hvilke grep må gjøres for å oppfylle lovverket?**

* Implementere brukervilkår og personvernserklæring.
* Sikre samtykke for datainnsamling og bruk av cookies.
* Gjennomføre regelmessige sikkerhetsrevisjoner for å beskytte brukerdata.

# Risikoanalyse og tiltaksplan

**Risikoanalyse**

* **Datasikkerhet:** Risiko for hacking og datainnbrudd.
* **Data Tap:** Risiko for tap av data grunnet tekniske feil.
* **Brukerautentisering:** Risiko for at uautoriserte brukere får tilgang.

**Tiltaksplan**

* **Backup-rutiner:** Implementere regelmessige backup-rutiner som beskrevet.

# Egenevaluering

**Har du vært effektiv**

Ja, prosjektet har fulgt tidsplanen og alle hovedmålene har blitt nådd innen de planlagte tidsfristene.

**Hva har du slitt med**

Jeg har slitt med komplekse nettverkskonfigurasjoner og multidimensjonale arrays, også AJAX som jeg har aldri jobbet med før.

**Hvor har du gjort det bra**

Jeg har gjort det bra med implementering av brukerregistrering, innlogging, og innleggssystemet.

**Hva har du blitt bedre på**

Jeg har blitt bedre på å bruke PHP for backend-utvikling og oppsett av virtualiserte miljøer ved hjelp av Hyper-V Manager.

**Hva ville du gjort annerledes**

Jeg ville ha investert mer tid i å lære nettverksadministrasjon og komplekse datastrukturer i JavaScript, men også prøve å bruke Python.

# Veiledning for lærlinger

Veiledning for Lærlinger: IMISHITPOSTSTATUS Website Setup  
  
Velkommen til veiledningen for lærlinger om oppsettet av nettsiden IMISHITPOSTSTATUS. Denne veiledningen gir deg en oversikt over hvordan nettsiden er strukturert og hvordan du kan samhandle med den.  
Innlogging og Opprettelse av Konto  
  
For å få tilgang til alle funksjonene på IMISHITPOSTSTATUS, må du først opprette en konto og logge inn. Klikk på "Login" i navigasjonsmenyen øverst på siden for å logge deg inn eller opprette en ny konto. Følg instruksjonene for å fullføre registreringen.  
Navigasjon  
  
Navigasjonen på nettsiden er enkel og intuitiv. Her er en oversikt over de tilgjengelige sidene:  
  
 Hjem: Dette er hovedsiden til nettsiden, hvor du kan finne en oversikt over de nyeste innleggene og aktivitetene.  
 Innlegg: Her kan du opprette og lese innlegg om forskjellige biler og diskutere med andre brukere.  
 FAQ: En side med vanlige spørsmål og svar om nettsiden og dens funksjoner.  
 Om Oss: Informasjon om IMISHITPOSTSTATUS-nettsiden og dens formål.  
 Kontakt: Kontaktinformasjon hvis du trenger hjelp eller har spørsmål.  
  
Opprettelse av Innlegg  
  
Når du er logget inn, kan du opprette nye innlegg på nettsiden. Gå til "Innlegg"-siden og klikk på "Opprett Innlegg"-knappen. Følg instruksjonene for å laste opp bilder og skrive en beskrivelse av bilen du ønsker å diskutere.  
Liking og Svar  
  
På hvert innlegg har du muligheten til å "like" innholdet og svare på det. Bruk "Like"-knappen for å vise din støtte eller beundring for et innlegg, og bruk "Svar"-knappen for å delta i samtalen og dele dine tanker og meninger.  
Teknologier Brukt  
  
Nettsiden er bygget ved hjelp av følgende teknologier:  
  
 Frontend:  
 HTML  
 CSS  
 JavaScript (JS)  
 JSX  
  
 Backend:  
 PHP 8.2  
  
 Database:  
 MYSQL  
  
Opphavsrett  
  
Alt innhold på nettsiden, inkludert tekst, bilder og annet media, tilhører IMISHITPOSTSTATUS med mindre annet er oppgitt. Uautorisert bruk eller reproduksjon av noe innhold er strengt forbudt.  
  
© 2024 IMISHITPOSTSTATUS. Alle rettigheter reservert.  
  
Takk for at du tar deg tid til å lære om oppsettet av IMISHITPOSTSTATUS-nettsiden. Hvis du har spørsmål eller trenger ytterligere hjelp, nøl ikke med å kontakte oss via kontaktsiden. Vi håper du nyter opplevelsen på IMISHITPOSTSTATUS!

# Mer om lovverk

Data Protection and Privacy Laws:  
  
 General Data Protection Regulation (GDPR):  
 Applicability: Applies to websites collecting personal data from users within the European Union (EU).  
 Requirements: Requires obtaining explicit consent for data collection, providing transparent privacy policies, implementing data security measures, and offering users rights regarding their personal data.  
  
 California Consumer Privacy Act (CCPA):  
 Applicability: Applies to websites collecting personal information from California residents.  
 Requirements: Mandates transparent privacy policies, opt-out options for data sharing/selling, and data breach notification obligations.  
  
Cybersecurity Laws:  
  
 Cybersecurity Information Sharing Act (CISA):  
 Applicability: Applies to organizations sharing cybersecurity threat information with the government or other entities.  
 Requirements: Encourages voluntary sharing of cybersecurity threat information for the protection of information systems.  
  
 Computer Fraud and Abuse Act (CFAA):  
 Applicability: Applies to unauthorized access to computer systems and networks.  
 Requirements: Prohibits unauthorized access to computer systems, networks, and data.  
  
Intellectual Property Laws:  
  
 Copyright Law:  
 Applicability: Protects original content, including website text, images, and design elements.  
 Requirements: Requires obtaining permission for the use of copyrighted materials and displaying copyright notices where applicable.  
  
 Trademark Law:  
 Applicability: Protects brand names, logos, and slogans associated with the website.  
 Requirements: Prohibits unauthorized use of trademarks and requires registration for trademark protection.  
  
E-Commerce Laws:  
  
 Electronic Commerce Directive (ECD) (EU):  
 Applicability: Applies to online service providers operating within the EU.  
 Requirements: Mandates disclosure of certain information (e.g., business details, pricing, terms and conditions) and establishes liability rules for intermediary service providers.  
  
 Electronic Transactions Act (ETA) (Various Countries):  
 Applicability: Governs electronic transactions and contracts.  
 Requirements: Recognizes the validity of electronic signatures and establishes rules for electronic contracts.  
  
Accessibility Laws:  
  
 Web Content Accessibility Guidelines (WCAG):  
 Applicability: Provides guidelines for making web content accessible to individuals with disabilities.  
 Requirements: Encourages compliance with accessibility standards to ensure equal access to web content for all users.

# Database

# Database Schema for IMISHITPOSTSTATUS

## Table: post\_replies

This table contains the replies to posts.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| post\_id | INT | NOT NULL | ID of the post being replied to |
| post\_reply | VARCHAR(255) | NOT NULL | Content of the reply |
| replied\_by | VARCHAR(255) | NOT NULL | Username of the replier |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each reply |

### Indexes

* posts\_id: Index on post\_id
* replied\_by: Index on replied\_by

## 

## Table: posts

This table contains the posts made by users.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each post |
| username | VARCHAR(255) | NOT NULL | Username of the user who created the post |
| post | VARCHAR(10000) | NOT NULL | Content of the post |
| likes | INT | DEFAULT 0 | Number of likes the post has received |
| carinfo | VARCHAR(255) | NOT NULL | Information about the car in the post |
| photo | LONGBLOB |  | Photo associated with the post |
| datetime | DATETIME | DEFAULT CURRENT\_TIMESTAMP, NOT NULL | Date and time when the post was created |
| dislikes | INT | DEFAULT 0 | Number of dislikes the post has received |

## 

## Table: user\_dislikes

This table tracks the dislikes made by users.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each dislike record |
| user\_id | INT | NOT NULL | ID of the user who disliked the post |
| post\_id | INT | NOT NULL | ID of the disliked post |
| disliked\_at | TIMESTAMP | DEFAULT CURRENT\_TIMESTAMP | Timestamp when the post was disliked |
| count | INT | DEFAULT 1, NOT NULL | Count of dislikes (usually 1) |

### Constraints

* unique\_like: Unique constraint on (user\_id, post\_id)
* user\_dislikes\_ibfk\_1: Foreign key referencing users(id)
* user\_dislikes\_ibfk\_2: Foreign key referencing posts(id)

### Indexes

* post\_id: Index on post\_id

## 

## Table: user\_friends

This table contains user friendship relations.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each friendship record |

## 

## Table: user\_likes

This table tracks the likes made by users.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each like record |
| user\_id | INT | NOT NULL | ID of the user who liked the post |
| post\_id | INT | NOT NULL | ID of the liked post |
| liked\_at | TIMESTAMP | DEFAULT CURRENT\_TIMESTAMP | Timestamp when the post was liked |

### Constraints

* unique\_like: Unique constraint on (user\_id, post\_id)
* user\_likes\_ibfk\_1: Foreign key referencing users(id)
* user\_likes\_ibfk\_2: Foreign key referencing posts(id)

### Indexes

* post\_id: Index on post\_id

## Table: user\_reports

This table contains user reports.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each report record |

## 

## Table: users

This table contains information about the users registered on the website.

| Column | Data Type | Constraints | Description |
| --- | --- | --- | --- |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for each user |
| username | VARCHAR(255) | NOT NULL | Username of the user |
| pwd | VARCHAR(255) | NOT NULL | Password for the user's account |
| email | VARCHAR(255) | NOT NULL | Email address of the user |
| admin | TINYINT(1) | DEFAULT 0, NOT NULL | Boolean flag indicating if the user is an admin |
| profile\_photo | LONGBLOB | DEFAULT 'profile.png' | Profile photo of the user |