# 

Jeep tur netside

Av Yosra Azzam

Table of contents

[0](#_Toc136633015)

[Innledning 2](#_Toc136633016)

[Prosjektplan / liste over oppgaver 4](#_Toc136633017)

[Dokumentasjon 5](#_Toc136633018)

[Netverkstegning 5](#_Toc136633019)

[Backup-rutiner 5](#_Toc136633020)

[database tilgang 6](#_Toc136633021)

[GitHub linken 6](#_Toc136633022)

[Fork repository av prosjektet 6](#_Toc136633023)

[Opplæringsvideo 7](#_Toc136633024)

[problemer som kom under utviklingen 7](#_Toc136633025)

[Kartlegging av lovverk 9](#_Toc136633026)

[Risikoanalyse og tiltaksplan 12](#_Toc136633027)

[Risikoanalyse 12](#_Toc136633028)

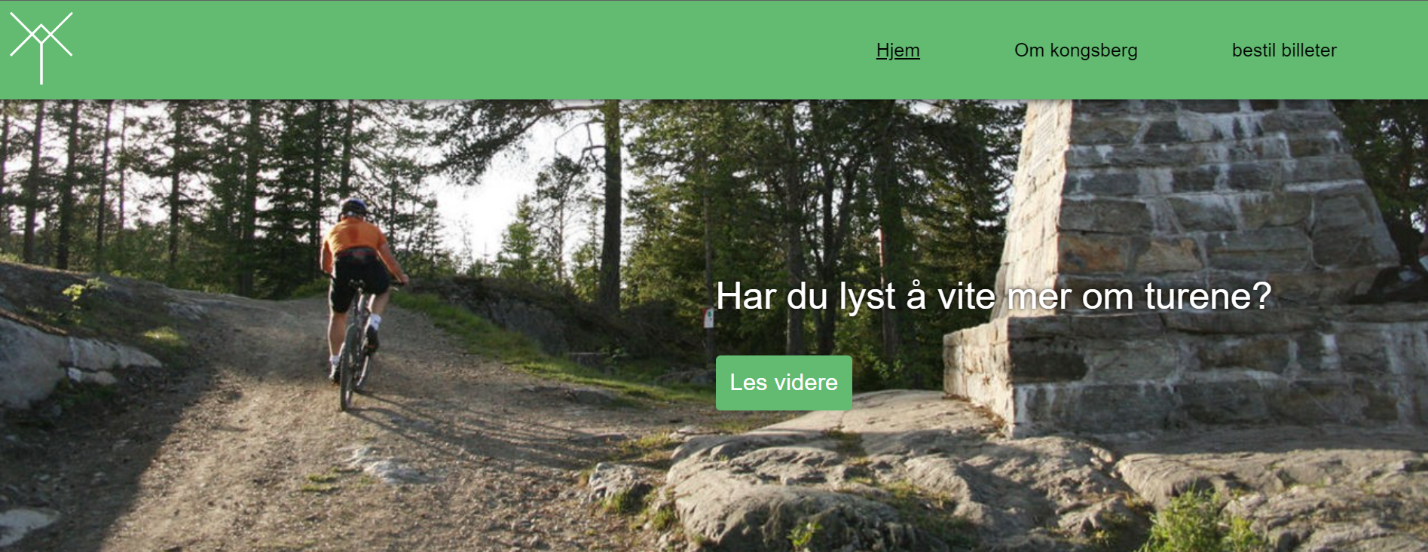
[Tiltaksplan 15](#_Toc136633029)

[Egenevaluering 16](#_Toc136633030)

[Kilder 17](#_Toc136633031)

# Innledning

Dette prosjektet handler om utviklingen av en nettside som formidler og promoterer Jeep-turer i Kongsberg. Nettsiden er designet for å gi potensielle kunder informasjon om turene som tilbys, og gir dem muligheten til å enkelt bestille og planlegge sin egen tur.

Gjennom bruk av HTML, CSS, JavaScript, PHP og MySQL har vi skapt en brukervennlig og funksjonell plattform som lar brukerne utforske ulike Jeep-turer, velge sine preferanser og fullføre bestillingen.

Målet med prosjektet er å tilby en brukervennlig nettside som gjør det enkelt for besøkende å finne relevant informasjon om Jeep-turer i Kongsberg, samt å bestille og tilpasse sine egne turer. jeg ønsket å skape en engasjerende opplevelse for brukerne ved å presentere relevant innhold, bilder og detaljer om turene, og samtidig sikre en sømløs bestillingsprosess.

A screenshot of a phone

Description automatically generated with medium confidence

Prosjektet har vært pågående i omtrent ett måned, og har involvert forskjellige stadier, inkludert planlegging, design, utvikling og testing. Nøkkeldelene i prosjektet har vært utviklingen av nettsidens front-end og back-end og samt sikkerhets- og databasemanagement-aspekter.

# Prosjektplan / liste over oppgaver

Jeg hadde ikke med en prosjekt plan siden det hjelper meg ikke, men jeg hadde en list av oppgaver som trengte å gjøres:

1. Planlegging
2. Identifisere prosjektmål og -omfang
3. Definere krav og funksjonalitet
4. Lage en tidsplan og ressursplan
5. Identifisere nødvendige verktøy og teknologier
6. Utvikling
7. Implementere HTML-struktur og CSS-styling
8. Legge til interaktivitet med JavaScript
9. Utvikle PHP-basert backend-funksjonalitet
10. Opprette og administrere MySQL-databasen
11. Implementere betalingsløsninger
12. Implementere brukerregistrering og -autentisering
13. Testing
14. Utføre enhetstesting av kodesnutter
15. Utføre integrasjonstesting av systemet
16. Identifisere og rette feil og mangler
17. Teste brukeropplevelsen og brukervennlighet
18. Utføre sikkerhetstesting og sørge for databaselagring
19. Server og database
20. Konfigurere servere og domenenavn
21. Flytte fra XAMPP til LAMP
22. Gjennomføre ytelsestesting og optimalisering
23. Dokumentasjon
24. Lage prosjektrapport
25. Lage brukerveiledning i form av film og dokument
26. Lage FAQ side

# Dokumentasjon

## Netverkstegning

A picture containing text, screenshot, diagram, design

Description automatically generated

## Backup-rutiner

* Regelmessige sikkerhetskopier av alle filer relatert til nettsiden, inkludert HTML-, CSS-, JavaScript- og PHP-filer. Dette kan gjøres manuelt ved å kopiere filene til et sikkerhetskopieringsområde eller ved hjelp av automatiserte verktøy for sikkerhetskopiering. GitHub er et nyttig verktøy til dette her, og man kan bruke skyen (i mitt tilfelle, OneDrive).
* Regelmessig kopiering av MySQL-databasen som inneholder bestillingsdata. Dette kan gjøres ved å eksportere databasen som en SQL-fil og lagre den i skyen.
* For å beskytte mot fysiske skader eller tap av data på stedet, er det anbefalt å lagre sikkerhetskopiene eksternt. Dette kan være gjennom bruk av eksterne harddisker, nettverkslagring (NAS) eller skybaserte lagringsløsninger. Det er viktig å sikre at sikkerhetskopiene er tilgjengelige og beskyttet mot uønsket tilgang.
* Regelmessig test gjenopprettingsprosessen ved å gjenopprette sikkerhetskopiene til et testmiljø. Dette vil bidra til å verifisere at sikkerhetskopiene er funksjonelle og at dataene kan gjenopprettes uten problemer.
* Opprett en oppdatert dokumentasjon som inneholder detaljer om backup-rutinene, inkludert frekvens, metoder og lagringssted.)
* Passord, brukernavn, porter etc (TBD)

## database tilgang

|  |  |  |  |
| --- | --- | --- | --- |
| Hostname | brukernavn | passord | Port |
| Localhost | root | yosraDB | 80 |

## GitHub linken

<https://github.com/Sunysian/eksamen>

## Fork repository av prosjektet

1. På GitHub.com, naviger til eksamen repositoryet.
2. Øverst til høyre på siden klikker du på «Fork».
3. Under «Owner» velger du rullegardinmenyen og klikker på en eier for forket repositoryet.
4. Som standard navngis forks på samme måte som oppstrømsrespositoyer. Hvis du vil skille forken ytterligere, skriver du inn et navn i feltet "Repository name".
5. Alternativt kan du skrive inn en beskrivelse av forken i feltet "Description".
6. Du kan også velge «Copy the DEFAULT branch only». For mange forkingscenarier, for eksempel å bidra til åpen kilde-prosjekter, trenger du bare å kopiere defult branchen. Hvis du ikke velger dette alternativet, kopieres alle branches til den nye forken.
7. Klikk på «Create fork».

## Opplæringsvideo

(kort video om nettsiden og hvordan man navigerer det)

<https://youtu.be/f59e8yxIgew>

## problemer som kom under utviklingen

* Det kan hennde at det var mer problemer, men alt jeg kan tenke på er database-konfigurasjon og problemer med koding som er ikke så viktig til å dukumentere.
* Hva jeg har lyst til å dokumentere er isteden er at jeg har på et tidspunkt slettet mitt tilgang til database (phpmyadmin) og en skjerm kom med «access denied».

A screenshot of a pink box

Description automatically generated with low confidence

* ChatGPT hjulpet her, og hva jeg gjort var å gå inni xampp mappe og åpnet en fil som heter *phpMyAdmin*, og så åpnet jeg en fil som heter *config.inc.php*. Der bytte jeg passordet til noe jeg har valgt fra før ved å finne dette linje «$cfg['Servers'][$i]['password'] = '';» Jeg putte inn mitt passord og det fungerte.

notater jeg hadde som egentlig er ikke så relevant. Dette var mitt gammel plan når siden var en eksamens oppgave, men tenkte det kan være greit å inkludere alt jeg har om prosjektet.

A screenshot of a computer program

Description automatically generated with low confidence

# Kartlegging av lovverk

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Lovverk** | **Hva handler dette lovverket om?** | **Hvilke paragrafer er relevante for mitt system?** | **Hvordan kan systemet risikere å bryte loven?** | **Hvilke konsekvenser kan det ha om systemet bryter denne loven?** | **Hva må jeg gjøre for å sørge for å ikke bryte denne loven?** |
| [Arbeidsmiljøloven](https://lovdata.no/dokument/NL/lov/2005-06-17-62?q=arbeidsmilj%C3%B8loven) | Arbeidsmiljøloven er en norsk lov som sikrer et trygt og godt arbeidsmiljø for arbeidstakere. Den regulerer arbeidsforhold, arbeidstid, pauser, hviletid og vern mot diskriminering og trakassering. Loven pålegger arbeidsgivere å sørge for sikre og helsemessig tilfredsstillende arbeidsforhold, samt å respektere arbeidstakernes rettigheter og beskyttelse. | Loven er ikke relevant til mitt prosjekt/system siden jeg er en eneste som jobber med det. | Siden loven er ikke relevant mitt prosjekt/system, så tenker jeg at det er ingen risiko for å bryte loven. | Konsekvensene av å bryte Arbeidsmiljøloven kan inkludere sanksjoner fra Arbeidstilsynet, erstatningskrav, arbeidsrettslige konsekvenser og reputasjonsskader for systemet. | Siden loven er ikke relevant, så bekymre meg ikke til at mitt prosjekt vil bryte loven. |
| [Forskrift om universell utforming av IKT-løsninger](https://lovdata.no/dokument/SF/forskrift/2013-06-21-732?q=universell%20utforming) | Forskrift om universell utforming av IKT-løsninger sikrer tilgjengelighet og universell utforming av digitale løsninger. Den gjelder for offentlige og private virksomheter og fastsetter krav til blant annet nettsteder og programvare. Målet er å sikre at alle, uavhengig av funksjonsnedsettelse, kan bruke digitale løsninger. | § 2. Virkeområde.  § 4. Krav til universell utforming av IKT  § 9. Tjenestebasert tilgjengelighetserklæring.  § 10. Krav til dokumentasjon og erklæring om universell utforming.  § 12. Konsekvensutredning og oppfølging. | Systemet kan risikere å bryte Forskrift om universell utforming av IKT-løsninger hvis det ikke oppfyller kravene til tilgjengelighet og universell utforming av IKT-løsninger. | Konsekvensene av å bryte Forskrift om universell utforming av IKT-løsninger kan inkludere rettslige sanksjoner, økonomiske bøter og krav om å rette opp manglene i systemet for å oppfylle kravene i forskriften. | Evaluer og identifiser mangler i universell utforming.  Implementer tilgjengelighetsstandarder som WCAG.  Tilrettelegg for tilgjengelig innhold og funksjonalitet.  Utfør regelmessige tilgjengelighetstester.  Opprett en offentlig tilgjengelighetserklæring.  Hold deg oppdatert og forbedre kontinuerlig. |
| [Person-opplysningsloven](https://lovdata.no/dokument/NL/lov/2018-06-15-38?q=personopplysning) | Personopplysningsloven (GDPR) er en norsk lov som regulerer behandlingen av personopplysninger og beskyttelsen av personvernet. Loven fastsetter prinsipper for behandlingen, krever samtykke, gir rettigheter til enkeltpersoner og setter krav til informasjonssikkerhet og overføring av data. Loven er viktig for å sikre personvern og må følges for å beskytte personopplysninger på riktig måte. | § 8. Grunnleggende prinsipper for behandling av personopplysninger:  § 9. Samtykke.  § 11. Rett til informasjon.  § 20. Sikring av personopplysninger.  § 28. Databehandleravtale. | Systemet kan risikere å bryte Personopplysningsloven hvis det ikke behandler personopplysninger i samsvar med lovens krav og prinsipper for behandling av personopplysninger. | Konsekvensene av å bryte Personopplysningsloven kan omfatte økonomiske bøter, omdømmetap, rettslige sanksjoner, erstatningskrav og pålegg om å rette opp i bruddene og endre behandlingspraksis. | Samle inn og behandle personopplysninger lovlig og på et rettmessig grunnlag.  Innhente gyldig samtykke fra de registrerte.  Beskytte personopplysninger gjennom sikkerhetsforanstaltninger.  Gi informasjon om hvordan personopplysninger behandles.  Respektere rettighetene til de registrerte, inkludert innsyn og retting av opplysninger.  Melde eventuelle brudd på personopplysningssikkerheten til tilsynsmyndigheter og berørte personer. |
| [GDPR](https://www.datatilsynet.no/regelverk-og-verktoy/lover-og-regler/om-personopplysningsloven-og-nar-den-gjelder/) | GDPR (General Data Protection Regulation) er en europeisk personvernlov som styrker beskyttelsen av personopplysninger. Den gjelder for organisasjoner som behandler personopplysninger om EU-borgere, uavhengig av hvor organisasjonen er lokalisert. Loven stiller krav til samtykke, gir enkeltpersoner rettigheter og krever informasjonssikkerhet. Organisasjoner må implementere tiltak for å oppfylle lovens krav og sikre ansvarlig behandling av personopplysninger. | Artikkel 5 - Prinsipper for behandling av personopplysninger.  Artikkel 6 - Lovlig behandling av personopplysninger.  Artikkel 13 - Informasjon som skal gis ved innsamling av personopplysninger.  Artikkel 25 - Data protection by design and by default.  Artikkel 32 - Sikkerhet for behandling av personopplysninger. | Systemet kan risikere å bryte GDPR hvis det ikke overholder de strenge kravene og prinsippene for behandling av personopplysninger som er fastsatt i loven. | Konsekvensene av å bryte GDPR kan innebære betydelige økonomiske bøter, omdømmetap, rettslige sanksjoner, erstatningskrav, pålegg om å rette opp i bruddene og midlertidig eller permanent stans av databehandlingsaktiviteter. | Behandle personopplysninger lovlig og på et rettmessig grunnlag.  Innhente og dokumentere gyldig samtykke for behandling av personopplysninger.  Sikre tilstrekkelig databeskyttelse og implementere passende sikkerhetstiltak.  Gi tydelig informasjon om behandlingen av personopplysninger.  Respektere rettighetene til de registrerte, inkludert innsyn, retting og sletting av opplysninger.  Melde eventuelle brudd på personopplysningssikkerheten til tilsynsmyndigheter og berørte personer. |

# Risikoanalyse og tiltaksplan

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Risikoanalyse | | | | | | | | | | |
| ID  nr | | **Aktivitet fra kartleggings-skjemaet** | **Mulig uønsket hendelse/ belastning** | **Vurdering**  **av sannsyn-**  **lighet** | **Vurdering av konsekvens:** | | | | **Risiko-verdi** | **Kommentarer/status**  **Forslag til tiltak** |
| (1-5) | Men-nesk-e  (A-E) | Ytre miljø  (A-E) | Øk/ mat-eriell  (A-E) | Om-døm-me  (A-E) |
| 1 | | Sikkerhetsbrudd og uautorisert tilgang til brukerdata. | Tap av personopplysninger, brudd på personvern. | 3 | D | A | C | E | D | Innføre sterke passordpolicyer og tofaktorautentisering. Implementere brannmurer og inntrengingsdeteksjonssystemer. Regelmessig overvåking av loggfiler for å oppdage uautoriserte aktiviteter. |
| 2 | | Teknisk svikt eller nedetid på nettsiden. | Tap av inntekt, redusert kundetilfredshet. | 3 | A | A | D | B | B | Implementere redundans og sikkerhetskopieringssystemer for å sikre tilgjengelighet. Overvåke nettsidens ytelse og tilgjengelighet kontinuerlig. Etablere en nødprosedyre for rask gjenoppretting av tjenesten ved nedetid. |
| 3 | | Manglende oppdatering av nettsiden og programvarekomponenter. | Sårbarheter for utnyttelse, sikkerhetsrisiko. | 2 | A | A | A | B | B | Etablere en oppdateringspolicy og rutiner for å holde nettsiden og programvarekomponenter oppdatert. Utføre regelmessige sikkerhetsoppdateringer og patching. Automatisere oppdateringsprosessen der det er mulig. |
| 4 | | Utilstrekkelig ytelse og skalerbarhet under høy belastning. | Nettsiden kan bli treg eller utilgjengelig for brukere. | 1 | A | A | B | A | A | Utføre grundige ytelsestester og lasttesting for å identifisere flaskehalser. Optimalisere koden og databasehåndteringen for bedre ytelse. Implementere caching-teknikker og ressursoptimalisering. |
|  |
|  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tiltaksplan | | | | |
| ID nr | **Tiltak** | **Ansvarlig** | **Tidsplan** | **Status** |
|
| 1 | Implementere sikkerhetstiltak som kryptering, brukerautentisering og tilgangskontroll for å beskytte brukerdata. | Utviklingsteamet (meg) | Gjennomføres før lansering av nettsiden. | Ikke iverksatt |
| 2 | Etablere rutiner for regelmessig sikkerhetsoppdatering av nettsiden og programvarekomponenter. | Utviklingsteamet (meg) | Gjennomføres før lansering av nettsiden og deretter kontinuerlig ved behov. | Ikke iverksatt |
| 3 | Implementere løsninger for systemovervåking og feilhåndtering for å raskt identifisere og håndtere tekniske problemer. | Utviklingsteamet (meg) | Gjennomføres før lansering av nettsiden. | Ikke iverksatt |
| 4 | Etablere backuprutiner og rutiner for gjenoppretting av data for å sikre at dataene kan bli gjenopprettet ved behov. | Utviklingsteamet (meg) | Gjennomføres før lansering av nettsiden. | Ikke iverksatt |

# Egenevaluering

Det var utfordringen med å bli overveldet av å prøve å lage et prosjekt som jeg skulle ha et år å lage (nærmere bestemt, 9 måneder), men i stedet hadde en måned å gjøre på grunn av ulike problemer. Det var den største utfordringen, å finne ut av alt i siste liten og ikke gjøre mitt beste arbeid. Alt ble forhastet.

Jeg har lært litt av java og python i starten da prosjektet mitt skulle være en visuell roman, jeg har lært mer om servere og databaser, php, mysql, lover, og listen fortsetter.

Når det gjelder å rangere arbeidet mitt, vil jeg si at jeg gjorde mitt beste med tanke på omstendighetene mine. Det er kanskje ikke opp til standardene til noen, men jeg er fornøyd med det jeg gjorde og holdt ut.

Det eneste jeg ville ha gjort annerledes er å sørge for at jeg undersøker språket/språkene jeg har tenkt å bruke for å gjøre oppgaven min. Å prøve å lære et nytt språk bremset meg betydelig, og det var alltid en risiko for at jeg ikke fant det raskt nok eller synes det var vanskelig å jobbe med, noe som skjedde to ganger. Det var dårlig dømmekraft fra min side, og jeg ville ha hatt det annerledes hvis jeg kjente risikoen bedre da.

# Kilder

W3schools

Stackoverflow

Youtube