**TICKETSYSTEM FOR BEDRIFT**

**PROSJEKTDOKUMENTASJON**

Fjellticket

TICKETSYSTEM FOR BEDRIFT

PROSJEKTDOKUMENTASJON

PROJECT DATE:

VERSION 1.0.0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PREPARED BY** | Erlend | **TITLE** | Project Manager | **DATE** | 16.05.24 |
| **APPROVED BY** | Erlend | **TITLE** | Project Manager | **DATE** | 16.05.24 |

Table of Contents

[MÅL MED PROSJEKTET 3](#_Toc166687045)

[PLANEN FOR PROSJEKTET 3](#_Toc166687046)

[Fase 1: Planlegging og design 3](#_Toc166687047)

[Fase 2: Implementering av backend (LAMP) 3](#_Toc166687048)

[Fase 3: Implementering av frontend (PHP) 3](#_Toc166687049)

[Fase 4: Testing og feilsøking 4](#_Toc166687050)

[Fase 5: Implementering og distribusjon 4](#_Toc166687051)

[PROGRAM OG INFRASTRUKTUR 5](#_Toc166687052)

[NETTVERKSDIAGRAM 5](#_Toc166687053)

[LOVER OG REGLER 6](#_Toc166687054)

[BRUKERE OG PASSORD 6](#_Toc166687055)

[VIDEO AV BRUK AV NETTSIDE 6](#_Toc166687056)

# MÅL MED PROSJEKTET

HVA DET FERDIGE PRODUKTET SKAL VÆRE

|  |
| --- |
| Et ticketsystem for ansatte der de kan sende in klager eller problemer.  Nettsiden skal være lett å forstå og enkel å bruke |

# PLANEN FOR PROSJEKTET

PLANEN FOR UTFØRING AV PROSJEKTET

|  |
| --- |
| Fase 1: Planlegging og design  1. **Kravspesifikasjon:**    * Identifiser kravene til ticketsystemet, inkludert funksjonalitet for å sende inn, vise og administrere klager.    * Definer brukerroller og tilgangsnivåer (administratorer (supportpersonell) og vanlige brukere). 2. **Database Design:**    * Design databaseskjemaet for å lagre klager, brukerinformasjon og andre relevante data.    * Bestem hvilke tabeller som trengs og å definere relasjonene mellom dem.  Fase 2: Implementering av backend (LAMP)  1. **Sett opp servermiljøet:**    * Installer og konfigurer Linux-operativsystemet.    * Installer Apache webserver, MySQL databaseserver og PHP-programmeringsspråk. 2. **Databaseimplementering:**    * Opprett databasen og de nødvendige tabellene i MySQL. 3. **Backend-logikk:**    * Utvikle PHP-skript for å behandle innsending av klager, lagre data i databasen og utføre nødvendig validering.    * Implementer autentisering og autorisasjon for brukerroller.  Fase 3: Implementering av frontend (PHP)  1. **Grensesnittdesign:**    * Design grensesnittet for ticketsystemet, inkludert skjemaer for innsending av klager, oversiktssider og administratorverktøy. 2. **Frontend-utvikling:**    * Bruk HTML, CSS og JavaScript for å implementere grensesnittet.    * Bruk PHP for å koble frontend til backend, for eksempel ved å sende skjema-data til serveren for behandling.  Fase 4: Testing og feilsøking  1. **Enhetstesting:**    * Test hver del av systemet separat for å sikre at det fungerer som forventet.    * Sjekk databaseoperasjoner, skjemainnsending, brukerinteraksjon og autentisering/autorisasjon. 2. **Integrasjonstesting:**    * Test hele systemet som en enhet for å sikre at alle deler fungerer godt sammen.    * Sjekk dataflyt mellom frontend og backend, brukeropplevelse og generell funksjonalitet.  Fase 5: Implementering og distribusjon  1. **Deploy:**    * Last opp filene til serveren via github og konfigurer alle nødvendige tillatelser. 2. **Opplæring og dokumentasjon:**    * Gi opplæring til brukere og administratorer om hvordan de skal bruke ticketsystemet.    * Lag dokumentasjon som beskriver systemets funksjonalitet, arkitektur og vedlikeholdsrutiner. |

## PROGRAM OG INFRASTRUKTUR

HVILKE PROGRAM SOM SKAL BRUKES

|  |
| --- |
| For skaping av prosjektet:   * Visual Studio Code * MYSQL Workbench   For server:   * Ubuntu (Linux) –Operativsystem * MariaDB (MYSQL) –Database * Apache –Webhost   For nettside:   * PHP * HTML |

## NETTVERKSDIAGRAM

DIAGRAM AV SAMMENKOBLINGER FRA SERVER TIL KLIENT

|  |
| --- |
|  |

# LOVER OG REGLER

HVILKE LOVER OG REGLER SOM ER TATT HENSYN TIL

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Lovverk | Hva handler dette lovverket om? | Hva må jeg gjøre for å sørge for å ikke bryte denne loven? | | Personvernloven | Behandlingen av personopplysninger | Ikke selge eller dele dataen jeg henter fra kunden, ha en sikker database. | | GDPR | Samme som personvernloven (EU lov ikke norsk lov) | -||- | | Ekomloven | Elektronisk kommunikasjon | Sørge for tilstrekkelig datasikkerhet, og om jeg skulle hatt nyhetsbrev måtte jeg spurt om samtykke. | | Åndsverksloven | Bruk av andres åndsverk | Ikke bruke bilder eller innhold som jeg ikke har lagd selv eller be om lov vis jeg skal. | |  |  |  |   Kilde: lovdata.no |

# BRUKERE OG PASSORD

|  |
| --- |
| På serveren:  Linux brukernavn….  Linux passord….  Mariadb brukernavn: root  Mariadb passord: root  På nettsiden:  Admin epost: [admin@admin.com](mailto:admin@admin.com)  Admin passord: admin |

# RISIKOANALYSE

|  |
| --- |
|  |

# RISIKOVURDERING

|  |
| --- |
|  |

# VIDEO AV BRUK AV NETTSIDE

|  |
| --- |
| En oppleringsvideo for administrerende brukere ligger i github prosjektet. |