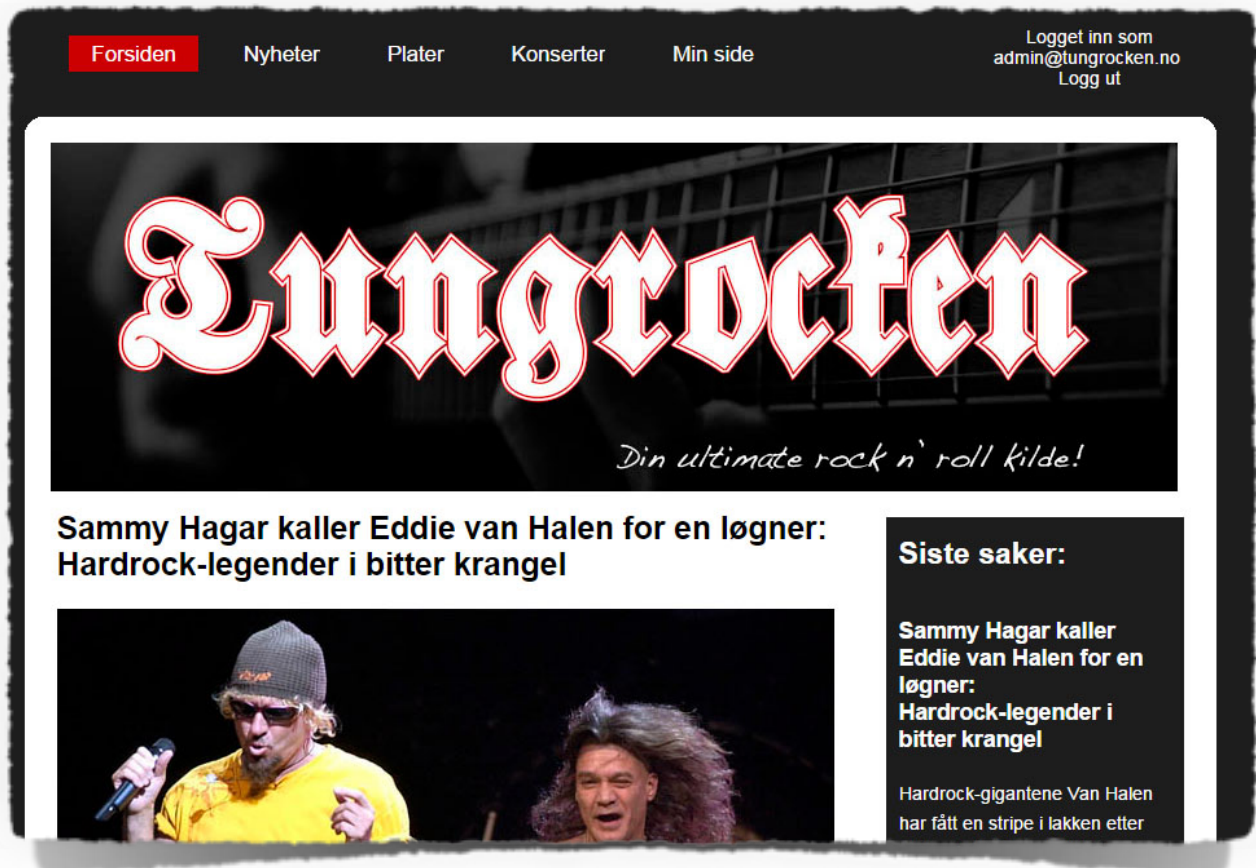


Musikkavisen

Tungrocken



Systemdokumentasjon

Gaute Hjelbakk Pettersen, Sindre Sjøholt,
Thomas Tennøy, Fredrik Mikael Valderhaug

LESERVEILEDNING, INNLEDNING

I denne systemdokumentasjonen blir det brukt en del uttrykk og begreper. Vi begynner med å klargjøre en del av disse.

Begrepsdefinisjoner

Database

En database er en strukturert samling av data, og begrepet er vanligvis knyttet til elektroniske systemer. Et databasesystem skal kunne behandle store mengder data effektivt og sikkert.

Databasespørring

En spørring brukes for å hente ut data fra en database. Kilden til disse dataene kan for eksempel være en tabell, en visning eller lignende.

Git

Et verktøy for versjonskontroll og kodedeling.

Publiseringssystem

Et publiseringssystem, innholdshåndteringssystem eller innholdsforvaltningssystem er et informasjonssystem for lagring, håndtering og organisering av innhold, og for publisering av innholdet på internett. Innholdet kan være av ulike typer. Det er som regel fleksibelt og utvidbart.

Open Source

Betyr «åpen kildekode», og at kildekoden til et dataprogram er gjort tilgjengelig (ofte på internett) for alle, som regel kostnadsfritt.

Forkortelser

SQL	Structured Query Language
PHP	Hypertext Preprocessor
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
XAMPP	Apache + MariaDB + PHP + Perl (en webserverløsning)

CMS Content Management System
URL Uniform Resource Locator

KRAVSPESIFIKASJON

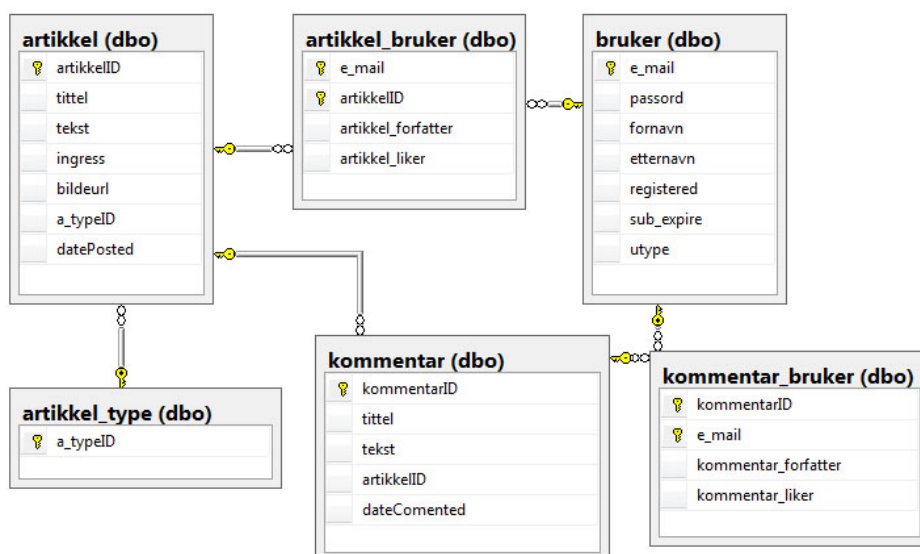
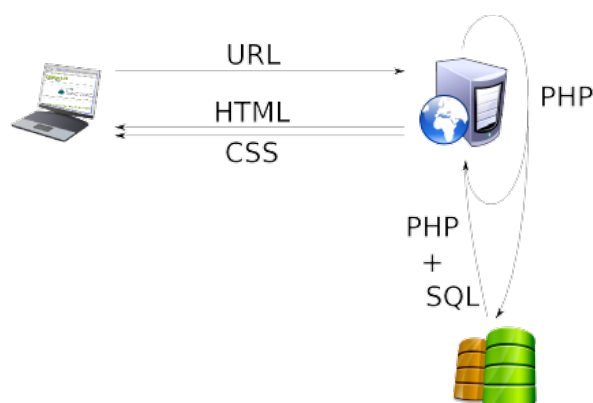
En tidligere utarbeidet kravspesifikasjon er vedlagt i sin helhet på slutten av dette dokumentet. Den er ikke endret eller forbedret, og eventuelle utbedringer og endringer i vårt system vil komme frem utifra denne (nye) rapporten.

HOVEDTREKKENE I KONSTRUKSJONEN

«Tungrocken» er en musikkavis på nett. Den er bygget opp som et CMS, eller publiseringssystem, der brukere med de riktige tilgangsrettighetene kan publisere eller endre innholdet på en nettside. I dette tilfellet nyhetsartikler, konsert- og plateanmeldelser.

Nettstedet er bygget opp med en database i bunnen, som kjører på systemet Microsoft SQL Server 2012. Nettsidene er konstruert ved hjelp av HTML, og bruker CSS for å ta seg av designutformingen. Nettsidene og databasen er knyttet sammen ved hjelp av PHP. Denne måten å bygge opp et nettsted på er velkjent og mye brukt, men i de fleste andre tilfeller vil små og mellomstore webbløsninger heller velge å benytte seg av MySQL-database i stedet for Microsoft sin som vi har brukt. Dette er trolig mye på grunn av at MySQL er «open source».

Videre så er selve databasen bygget opp



slik som modellen her viser. Dette er stort sett den samme modellen som vi har jobbet ut i fra siden begynnelsen, men noen små endringer underveis. Slik det er nå, så er ikke alle tabellene i bruk, men er beholdt med tanke på senere utvidelse av funksjonalitet og videre utvikling. Dette gjelder spesielt delen med å telle «liker-klikk» på artikler og kommentarer.

FUNKSJONELL BESKRIVELSE

Brukerne på dette nettstedet blir delt inn i fire kategorier. Det er «gjest», «betalende bruker», «journalist» og «administrator». Disse skiller seg fra hverandre på hva de har tilgang til.

Gjest

En gjest er hvem som helst som ikke er innlogget, altså ikke en registrert bruker. Denne typen bruker har tilgang til fremsiden på nettstedet, der det blir fremvist små smakebiter av de siste artiklene. Det er ikke mulighet for å se hele artikkelen uten at brukeren er pålogget.

Betalende bruker (abonnent)

En betalende bruker har full lesetilgang på alle artikler, så lenge den er innlogget. Brukeren har også tilgang til å både lese og skrive i kommentarfeltet under hver artikkel. Brukeren har tilgang til å gjøre små endringer, som skifte passord, under «Min side». En vil her ellers få informasjon om sitt abonnement. Et vanlig abonnement varer i 12 måneder.

Journalist

Journalisten er en registrert bruker uten noe satt dato for når brukeren utgår, slik det er for en abonnent. Journalisten har både lese og skriverettigheter, og er den brukertypen som skal styre innholdet på nettsiden. Legge ut nye artikler, samt slette eksisterende.

Administrator

Denne brukeren har alle rettigheter. Er den som skal styre det hele, og passe på at alt er som det skal være, både når det gjelder innholdet på nettsidene, men også når det kommer til



Logg inn

Epost/Brukernavn:

Passord:

[Logg inn](#) [Nullstill](#)

brukerkontroll. Denne brukeren blir satt opp av databaseadministratoren.

Nettsiden i bruk

Som tidligere nevnt, så vil en gjest møte en side med smakebiter på nyheter og andre saker. Så lenge brukeren ikke er innlogget som en av de tre andre typene bruker, er dette alt en vil få. Ved forsøk på å gå videre (inn på artikkel), så vil en bare bli sendt til skjema for innlogging.

Registrerte brukere av alle typer vil etter innlogging av tilgang på de respektive nivåer. En abonnent vil da kunne lese alt av innhold, samt benytte seg av kommentarfeltet.

Sidene er bygget opp slik at

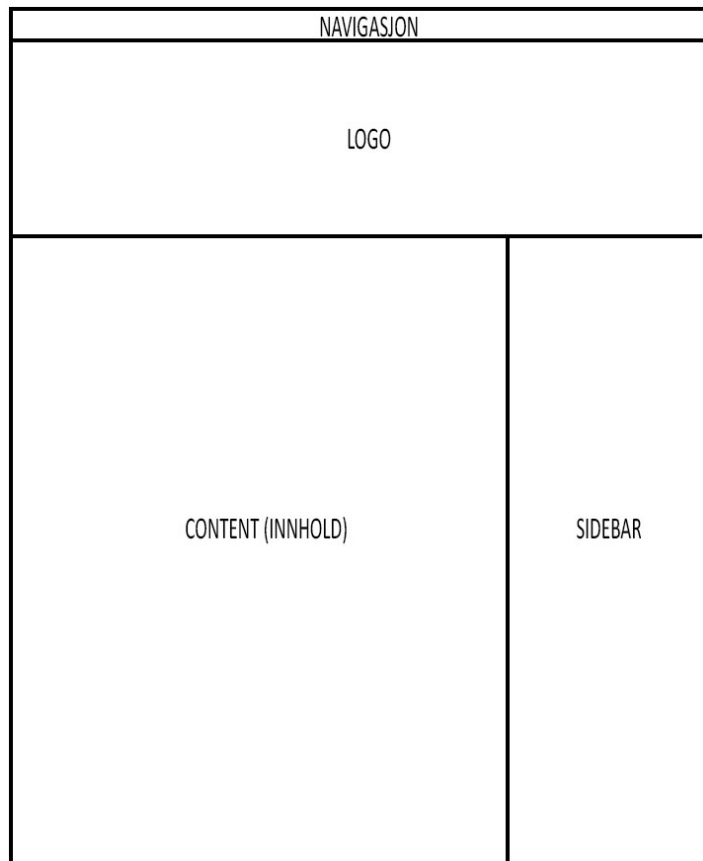
artikkelen bruker store deler av det feltet vi kaller «content» (se grafisk oppsett), med kommentarfeltet under selve artikkelen. I feltet «sidebar» vil det til en hver tid bli vist en liten smakebit av de fire siste sakene som er publisert, uavhengig av kategori.

Helt øverst finner vi navigasjonen, som brukeren kan benytte til å komme seg rundt på nettsidene. Her finner vi «Forsiden», «Nyheter», «Konserter», «Plater» og «Min side». Sistnevnte er brukeren sin egen side, og informasjonen her er generert ut i fra hvem som er pålogget. Innholdet her er kun tilgjengelig for den som er pålogget, og kan blant annet være informasjon om passord og utløpsdato på abonnementet.

Øverst i høyre hjørne finner vi knapp for «Logg inn» og

«Registrere» dersom brukeren er en gjest. Om brukeren er innlogget, står det hvem som er innlogget på systemet, samt en link for utlogging. Brukerregistrering foregår på et enkelt skjema, vist i bilde til høyre.

Helt nederst, under kommentardelen, finnes det link til kontaktinformasjon, samt en link som påloggede administratorer og journalister bruker for å komme til kontrollpanelet. Dette



Registrering av ny bruker:

Brukernavn (epost-adresse):

Passord:

Fornavn:

Etternavn:

kontrollpanelet inneholder meny for å skrive artikler, slette artikler, samt brukerkontroll for den som har tilgang til det.

YTRE GRENSESNITT OG INDRE LØSNINGER

Som tidligere nevnt, så bruker vi en database av type Microsoft SQL (MSQL), som er knytt sammen med vår nettside ved hjelp av programmeringsspråket PHP. All kommunikasjon mot databasen skjer med PHP. Vi har bygget nettsidene opp i separate deler, for enklere å kunne oppdatere eller endre funksjoner. Dette er spesielt nyttig for funksjoner som blir brukt flere steder. En slik funksjon er selve oppkoblingen mot databasen. Dette foregår i et eget script, som blir kallet på av de dokumentene som behøver kontakt med databasen.

```
<?php
$serverName = "(local)"; // eller "(local)"
$connectionInfo = array( "Database"=>"musikkavis", "UID"=>"user", "PWD"=>"password", "CharacterSet"=>"UTF-8" );
$conn = sqlsrv_connect( $serverName, $connectionInfo );
if( $conn === false ) {
    die( print_r( sqlsrv_errors(), true));
}
?>
```

Det blir gjort et forsøk på å koble seg opp mot databasen med serveradresse, navn på basen, brukernavn og passord. Feilmelding vil bli returnert om noe skulle gå galt.

Session

Videre så bruker vi et såkalt session-script, eller økt-script. Dette passer på at når brukeren først er logget inn, så slipper han å gjøre det for hver side han går inn på. Nettleseren sjekker hele tiden om at økten fremdeles pågår, og holder brukeren innlogget til han selv velger å logge seg av. Dette scriptet holder også kontroll over hvilken type bruker den innloggede er.

Henting av innhold

Når det kommer til selve uthenting av data (i vårt tilfelle artikler og bildelinker) fra basen, foregår det på separate SQL-spøringer i de respektive tilfellene data trengs. Om en artikkel skal hentes, så gjør vi det ved hjelp av en artikkel-ID i databasen. Scriptet/spørningen tar med seg en ID fra URL'en, f.eks. /article.php?id=23. Da

```
global $conn;
$id = $_GET["id"];

$stmt = "SELECT * FROM artikkel where artikkelID = $id";

$stmt = sqlsrv_query( $conn, $stmt );
```

blir tallet 23 tatt med inn i spørringen mot databasen, for å hente ut ønsket artikkel. En spørring etter artikkel er vist på bildet.

Måten innholdet vises på fremsiden, er litt spesielt i forhold til de typiske artikkelsidene. Dette fordi at det her bare skal være smakebiter av innholdet, for å få brukeren til å registrere seg for å få mere. Derfor vises bare et artikkelbilde, samt ingressen tittel og ingress til saken for å vise hva det dreier seg om. Siden er bygget opp slik at den nyeste saken vises i full bredde, mens de resterende sakene vises i halv størrelse i en dobbel kolonne under «toppsaken». For å få til dette blar vi gjennom hele databasen og fremviser en sak etter den andre, og så styrer vi fremvisningen ved hjelp av litt finurlig PHP-koding sammen med CSS. Det vi henter ut fra databasen blir blant annet lagt i et array (en samling) som vi jobber oss gjennom etterpå. Noe av koden er fremvist i bildet høyre.

```
while( $row = sqldr_fetch_array( $fget, SQLSRV_FETCH_ASSOC)){  
  
    $articleID = $row['artikkelID']; //sqldr_get_field( $stmt, 0 );  
    $tittel = $row['tittel']; //sqldr_get_field( $stmt, 1 );  
    $tekst = $row['tekst']; //sqldr_get_field( $stmt, 2 );  
    $ingress = $row['ingress']; //sqldr_get_field( $stmt, 3 );  
    $bildeURL = $row['bildeurl']; //sqldr_get_field( $stmt, 4 );  
  
    if ( $tittel === false ) {  
        die( print_r( sqldr_errors(), true ));  
    }  
}
```

Den grafiske effekten på navigasjonsmenyen er skapt ved hjelp av et JQuery Javascript med navn Lava Lamp.

Grafisk brukergrensesnitt

Temaet for nettstedet er som kjent tungrock. Eller hardrock, metall eller heavy rock om du vil. Fargevalgene vi har tatt i forhold til designet er derfor mye sort og hvitt, blandet med nyanser av grå og detaljer i rødt for å skape et klassisk tungrock-uttrykk. Fordelene med å bruke denne «fargepaletten» er flere. Blant annet så kan vi i mye mindre grad tenke på problemstillinger rundt fargeblindhet. Når hovedfargene er sort, hvitt og grått, så vil ikke dette skape nevneverdige problemer for de mest utbredte typene av fargeblindhet:

Protanopi - rødblindhet, 1 % av alle menn

Deutanopi - grønnblindhet, 1 % av alle menn

Tritanopi - blåblindhet, sjelden

Akromatopsi - fullstendig fargeblindhet, sjelden

Når det kommer til valget med å bruke sort tekst på hvit bakgrunn i artiklene, så er det den mest benyttede kombinasjonen. Den gir god kontrast, og er dokumentert den som er lettest å lese (*Bauer og Cavanaugh (1980) fant ut at deltagere i en lesetest fikk med seg 26% mere når de leste mørk tekst på lys bakgrunn*).

ENDRINGER OG UTVIDELSER

I forhold til den tidligere lanserte Kravspesifikasjonen, så er det ikke blitt gjort noen vesentlige endringer i applikasjonen. Vi har ikke «likerklikk» på artikler og kommentarer slik som planen var, men det foreligger muligheter for å implementere dette senere. Databasen er klargjort for dette.

Ellers så er applikasjonen produsert med den funksjonaliteten som var tiltenkt fra begynnelsen av.

Når det kommer til muligheter for videreutvikling og utbygging, så foreligger det ingen konkrete planer om dette på nåværende tidspunkt. Mulighetene for dette er selvsagt tilstede.

REFERANSER OG KILDER

Databasesystemer, 3. utgave // Bjørn Kristoffersen. Universitetsforlaget.

Datamodellering - praksis og teori // Edgar Bostrøm. MetodeData as.

Designing with the mind in mind // Jeff Johnson. Morgan Kaufmann.

Wikipedia.org // Diverse artikler.

PHP.net // Diverse artikler

W3schools.com // Diverse artikler

Msdn.microsoft.com // Diverse artikler

Kravspesifikasjon

Musikkavisen «Tungrocken»

FORMÅL VED KRAVSPESIFIKASJONEN

Følgende kravspesifikasjons formål er å gi en overordnet beskrivelse over utviklingen av vårt produkt; i dette inngår spesifikasjoner av utviklingsmetode, formål ved produktet, samt begreper, tekniske spesifikasjoner og krav for produktet.

UTVIKLING AV PRODUKT

I dette prosjektet har vi brukt SQL Server 2012 for å lage en database som skal holde informasjonen som nettsiden trenger for å fungere slik vi ønsker. Vi har brukt HTML og CSS for å lage nettsiden og PHP for å «snakke» mellom databasen og nettsiden. Vi har brukt programvaren «Git» og informasjonsbrønnen «github.com» for å kollaborere på prosjektet, slik at alle utviklerne har de oppdaterte filene i prosjektet til enhver tid.

Vi startet med å lage en database som vi så for oss at en slik nettavis måtte ha. Deretter opprettet vi en webserver, og brukte mye tid på hvordan vi skulle få databasen og webserveren til å snakke sammen. Etter mye prøving og feiling, fikk vi serveren og databasen til å snakke sammen med PHP-script.

Etter dette utviklet vi nettsidens utseende og funksjonalitet, slik at interaksjonen med database og server ville gå smertefritt. Som webserver brukte vi programmet XAMPP, som kjører på Apache-server.

APPLIKASJONENS FORMÅL

Produktets formål er å tilby brukere en nettside der de kan tilegne seg informasjon om musikk av genren tungrock. På nettsiden vil det bli lansert artikler om artister og band. Det vil hovedsaklig være konsertanmeldelser og annen informasjon om artistene. Produktet er tilpasset brukere som selv ønsker å oppdrive informasjon om artister innen sjangeren tungrock.

Applikasjon vil være bygd opp slik at den har fire typer brukere. Administrator, journalist, betalende bruker og gjest(ikke betalende bruker). Der Administratoren har det overordnede ansvaret, der journalister har skrive og leserettigheter, dvs. at de kan skrive innhold og redigere innhold. Betalende brukere har leserettigheter, og kan kommentere innhold. De vil også ha mulighet til å «like» innhold eller andres kommentarer. Gjesten vil kun ha mulighet til å se deler av innholdet på nettsiden. Denne typen brukere vil ikke ha noen andre rettigheter. For å få leserettigheter må brukerne betale,

slik at de kan se hele innholdet på nettsiden. Da får de et abonnement som kan fornyes, når det går ut.

APPLIKASJONENS FUNKSJONER

Presentere artikler med nyheter, konsertanmeldelser osv. Det vil være et kommentarfelt under artiklene, slik at brukerne kan bidra i diskusjonen, komme med tilleggsinformasjon, synspunkter eller kritikk. Det vil være mulig å gi respons både på artikler og kommentarer med såkalte «likerklikk».

Adminbrukeregenskaper:

- Administrere brukerrettigheter
- Overordnet ansvar
- Alle rettigheter

Journalistbrukeregenskaper:

- Skrive og leserettigheter
- Ansvar for publisering og vedlikehold av innhold

Betalende brukeregenskaper:

- Leserettigheter
- Kommentering av innhold
- «Liker» funksjon på innhold, og på kommentarer som er skrevet om innhold

Ikke betalende (gjest) brukeregenskaper:

- Leserettigheter på deler av innholdet

TEKNISKE DEFINISJONER, ACRONYMER OG FORTKORTELSER

HTML: (HyperText Markup Language) er et markeringsspråk for formatering av nettsider med hypertekst og annen informasjon som kan vises i en nettleser.

CSS (Cascading Style Sheets) er et språk som brukes til å definere utseende på filer skrevet i HTML eller XML.

PHP (Hypertext Preprocessor) er et dynamisk, tolket og løst typet programmeringsspråk hovedsaklig brukt for å utvikle dynamiske nettsider.

XXAMP er en gratis og «open source cross-platform web server solution pack» pakke utviklet av Apache Friends.