

**Emnekode: DS3111**

**Emnenavn: Content Management Systems**

**Vurderingskombinasjon: Skriftlig individuell hjemmeeksamen**

**Utleverings- og innleveringsfrist: 1. oktober–11. november**

**Filformat: .zip-fil**

## Eksamen

### 1. 1 til 3 studenter i hver gruppe

### 2. Obligatorisk presentasjon

- a. Muntlig presentasjon foran klassen, vise frem konsept og prototyper for å motta tilbakemelding for videre utvikling
- b. Dette skal gjennomføres mellom uke 5 og 9
- c. Grupper velger når de vil presentere, og det er forventet at framgangen gjenspeiler hva som har blitt dekket i forelesningene

### 3. Prosjekt

- a. Nettside utviklet i WordPress
  - i. Vektes 50% av karakter
  - ii. Hva som er kravene er beskrevet som 8 oppgaver under
- b. Skriftlig rapport
  - i. Vektes 50% av karakter
  - ii. Dokumentere de 8 oppgavene for utviklingen av nettsiden

### 4. Levering

- a. Gruppene skal laste opp til WiseFlow:
  - i. Skriftlig rapport
  - ii. En zip-fil med systemfiler (beskrevet i slutten av dokumentet)
  - iii. YouTube video (skjermopptak, beskrevet i slutten av dokumentet)

### Oppgave 1 – Konsept og prototype

Beskriv konseptet dere har valgt. Hva slags løsning er dette, og hva slags behov skal den dekke? Husk å både nevne behovene til eier av siden, og den faktiske sluttbrukeren av nettstedet. Beskriv dette som "Use Cases" eller "User Stories" for å vise frem i YouTube videoen.

Når alt over er klarlagt bør man utvikle en prototype (ofte lager man flere) for å få en generell oversikt over hvor man ønsker å ende opp.

Prototyping kan gjøres på forskjellige måter – f.eks. med mockups:

- Bilder (Photoshop, e.l.).
- Wireframe (en annen form for bilder).

- Interactive wireframe: Balsamiq (<http://balsamiq.com/products/mockups/>).
- HTML/CSS - kjapp (håndkodet) site for å finne ut av struktur/layout.

Visuelle mockups kan detaljeres så mye at man ender opp med en god oversikt over struktur, innholdstyper, layout, m.m. Velger man å gjøre dette nøye før man starter implementeringen, sparer man ofte tid senere i prosessen.

Skriv ned konseptet, inklusive prototype (legg den ved om mulig), i følgedokumentet.

## **Oppgave 2 – Klargjøring av default WordPress installasjon**

Det forutsettes at WordPress er installert.

Gjør forbedringer i konfigurasjonsfilene til installasjonen. Tenk sikkerhet og forenkling av kommende websiteutvikling.

Gjør endringer i WordPress oppsettet (Dashboard) som styrker deres løsningskonsept. Blant annet for: (ikke nødvendigvis begrenset til)

- General
- Writing
- Reading
- Discussion
- Media
- Permalinks
- User roles

Noter ned hvilke forbedringer dere gjør i følgedokumentet. (Eks: "Oppgave 2: Valgte å endre ... til ... da dette gjør at ...")

## **Oppgave 3 – Theme selection**

Theme fra nettet bør styres av hvilket innhold som skal vises og hvordan de som skal bruke løsningen etterpå vil arbeide med publisering av innhold.

Minimum bør et theme man velger fra nettet ha:

- Enkel og oversiktlig navigasjon.

- God top level access, få undermenyer om mulig.
- Design/layout som matcher (og ikke overdøver) innhold.
- Mobil/device tilpassede sider.
- Følge webstandarder mest mulig.
- Fungere likt på flest mulig browsere.

Velg et theme som aktivt følges opp (utvikles) og har tilgjengelig support. Det er lurt å bruke litt tid på å velge riktig "første gangen", så man ikke må gjøre mye arbeid om igjen senere.

Noter ned kriteriene for deres valg av theme i følgedokumentet. Kommenter også fremgangsmåten dere jobbet med child theme'et.

**Merk: Oppgaven kan løses på to måter, gruppene velger selv:**

1. Lag et "child theme" basert på valgt theme fra nettet
2. Bygg et theme "fra bunnen av"
  - a. Med "fra bunnen av" menes ikke å begynne med en tom wp-content/themes/xxx mappe. Det menes å begynne med en theme template som aktivt utvikles og som bruker nyeste webstandarder som utgangspunkt, f.eks.: <http://underscores.me>. (Utvikles og vedlikeholdes av de som lager WordPress.)

**NB: Dere skal IKKE lage 2 ferdige themes! Dere skal laste ned et godt utgangspunkt ELLER dere skal starte på en "fra bunnen av". I praksis så kommer dere til ha en flere themes i den lokale installasjonen deres, men dere skal bare ferdigstille og levere en.**

#### **Oppgave 4 – Plugin & widget selection**

Vis at du behersker plugins og widgets, ved å ta i bruk et knippe av de for din løsning.

Har du en klar oppfatning av hva slags funksjonalitet du ønsker gjennom plugins og widgets, finn komponenter som løser akkurat dette.

Ønsker du innspill til nyttige plugins og widgets? Her er noen forslag:

#### **Plugins**

- Caching
- Security enhancements (login limiter)
- Statistics
- Social Sharing

### Widgets

- Search
- Latest posts
- Latest comments
- Social feeds
- Social sharing/activities

Noter ned i følgedokumentet hvilke plugins dere så behov for, og hvorfor dere valgte de dere gjorde.

### Oppgave 5 – Produksjon av tekst og bilder

Med utgangspunkt i kunnskapen dere har fått knyttet til produksjon og vedlikehold av innholdet til et CMS, sørg for at deres tekst og bilder følger disse reglene/standardene.

Dere skal i tillegg i følgedokumentet:

1. Dokumentere og begrunne de valgene dere har tatt i forhold til bruk av tekst og bilder i løsningen som er utviklet, og:
2. Redegjøre for og komme med forslag til hvordan innholdet (tekst og bilder) skal holdes kontinuerlig oppdatert etter at løsningen er lansert.

### Oppgave 6 – Universell utforming

I denne oppgaven skal dere gjøre rede for hvordan dere vil sikre at deres ferdigutviklede nettsted i størst mulig grad oppfyller kravene til universell utforming av nettsteder.

Dere bør her ha med både hvordan dere vil sikre dette i designfasen, og hvordan dere vil teste løsningen underveis og når den er ferdigstilt. Hint: listen over praktiske tips fra forelesningen i uke 37 bør brukes for å teste løsningen. Husk å dokumentere dette til slutt.

### Oppgave 7 – PHP tilpasninger

Vis at du behersker PHP for WordPress, ved å gjøre minst én av følgende:

- Gjør endringer i PHP koden til eksisterende theme, plugin eller widget.
- Skriv en ny plugin eller widget for å løse et behov der du ikke har funnet noen god, eksisterende komponent. (NB: Det finnes plugins for nesten alt man kan tenke seg, så kanskje må man lage noe som alt eksisterer – det er i så fall ok for denne oppgaven.) Tips til dette punktet – mulige PHP tilpasninger:

**Lette**

- Theme – functions.php:
  - Lag og legg til CSS for print, web fonts og editor-style, eller
  - Legg til støtte for en eller flere egendefinerte image sizes med `add_image_size()`:  
[http://codex.wordpress.org/Function\\_Reference/add\\_image\\_size](http://codex.wordpress.org/Function_Reference/add_image_size)
- Theme – templates:
  - Bruk egendefinerte image sizes i header, footer, single etc., eller
  - Lag og legg inn JavaScript web tracking code for Google Analytics og verifiser at dette fungerer som forventet (på localhost). Dette krever justering av web tracking code: [Devguide](#) eller [Stackoverflow info](#)

### Middels

- Lag en plugin som generer Open Graph meta tags for deling av sider/innhold på Facebook:  
<https://developers.facebook.com/docs/sharing/best-practices#tags>

### Vanskelig

- Lag en plugin som viser de 5 mest leste sakene (most read posts) for de siste 14 dagene med featured image thumbnail, overskrift og kort uttrekk av brødtekst per sak. Eksempel: Footer på [www.kunstforum.as](http://www.kunstforum.as). En måte å gjøre dette på er å bruke WordPress.com Stats plug-in eller Google Analytics som datagrunnlag.

Lag en kort teknisk dokumentasjon/brukermanual for elementer dere lager fra bunnen av eller tilpasser utenom det vanlige.

Kommenter i følgedokumentet hvilke themes, plugins og/eller widgets du har endret/skrevet fra bunnen av, samt hvordan og hensikt.

### Oppgave 8 – Ferdigstilling

Gjennomfør testing av løsningen. Når du er ferdig med dette punktet, skal løsningen være klar for **levering**. Har testingen avslørt mangler, skal disse forsøkes å fikse, noter ned disse og beskriv hvordan de ble fikset eller grunnen for at de ikke kunne bli fikset.

Beskriv testmetodene og -prosessen i følgedokumentet. Lag samtidig en fiktiv plan for backup og vedlikehold av løsningen. ("Fiktiv" som i at du ikke trenger å gjennomføre fremtidig vedlikehold i praksis.)

Ferdigstill samtidig dokumentet, med hensikt at det skal leveres inn i WiseFlow og er vektet 50% av eksamenskarakter.

**Levering**

Løsningen skal også leveres i WiseFlow, som en zip-fil. Du finner det du trenger i "htdocs" og "wp-content" mappene. Det som skal inkluderes i zip-filen er:

- Mappen med ferdigstilt tema, i en mappe som heter "themes"
- Mappen(e) med plugins, i en mappe som heter "plugins"
- Wp-config.php og bitnami\_wordpress.sql

I tillegg så skal det lages og lastes opp en YouTube video der 3 use cases/user stories blir presentert fra design til implementasjon. Lenken skal skrives i rapporten, sammen med admin brukernavn og passord til WP Dashboard (admin panelet) til løsningen din.

**- Slutt på oppgavesettet -**