

# Manual för Vänsterliberalernas hemsida

Togald Nilsson

21 mars 2013

## Innehåll

<b>1</b>	<b>Inledning</b>	<b>2</b>
<b>2</b>	<b>Hur fungerar det?</b>	<b>2</b>
2.1	Struktur i <code>content/</code> . . . . .	2
2.2	Arbetsflödet . . . . .	3
2.3	PHP-variabler du kan använda . . . . .	3
2.4	Att använda PHP dynamiskt på servern . . . . .	4

---

## 1 Inledning

Vänsterliberalernas hemsida är inte som vilken hemsida som helst. Hemsidan skapas med en teknik som kan kallas developer-side PHP, vilket innebär att hemsidan genereras med hjälp av PHP av utvecklaren, innan den läggs upp på servern. Man kan säga att webbsidan måste kompileras innan den kan användas. PHP används av den anledningen att det är lätt att bädda in i HTML-koden.

Eftersom webbsidan i dagsläget bara använder PHP på utvecklarsidan, behövs ingen server med PHP installerat för att köra webbsidan. Däremot måste utvecklaren ha PHP installerat på sin dator. Det är möjligt att använda PHP för att generera dynamiskt innehåll på webbsidan också, mer om det i sektion 2.4.

## 2 Hur fungerar det?

Vänsterliberalernas hemsida har tre mappar, `content/`, `source/` och `target/`.

- `content/` innehåller mallar till alla de sidor som ska genereras.
- `source/` innehåller alla källfiler, `creator.php`, `skel.php` och bilder, style sheets och så vidare som ska finnas på sidan.
- `target/` är vanligtvis tom vid start. Det är i denna mapp som webbsidan genereras.

`source/creator.php` är den fil där all magi sker. Det är den som används för att kompilera mallarna till en fungerande webbsida. Filen är väl kommenterad, och dess arbetsordning beskrivs här.

1. Leta igenom `content/` efter mallar, och kompilera en webbsida för varje mall som hittas med `source/skel.php` som skelett.
2. Placera mallfilen på motsvarande plats i `target/`. Alla filer som inte är indexfiler kopieras utan behandling.
3. Kopiera över bilder, style sheets och andra statiska dokument från `source/` till `target/`.

### 2.1 Struktur i `content/`

För varje webbsida som ska skapas, måste en mapp med sidans namn skapas i `content/`. I denna mapp skapas en fil `index.php`, som är den fil som används av `source/creator.php` för att generera webbsidan. Indexfilen behöver inte

---

innehålla någon struktur alls, förutom en deklaration av PHP-variabeln `$title` någonstans i dokumentet, förslagsvis i början. `$title` används för att skriva ut sidans titel i titelraden i webbläsaren.

Fördelarna med att använda mappar istället för HTML-sidor för att definiera strukturen, är framför allt att

- URL-erna blir enklare, till exempel `http://v1.mfserver.net/politik/` istället för `http://v1.mfserver.net/politik.php`. Detta är viktigt framför allt för undersidor, som annars kan få mycket skumma URL-er.
- det gör det lättare att skapa undersidor till webbsidorna. Istället för att skapa en separat mapp för undersidor till en viss sida, lägger man bara alltsammans i sidans mapp.

## 2.2 Arbetsflödet

1. Kлона webbsidan från GitHub. Du borde veta hur man gör det. Tips: `git clone`. Om du redan har webbsidan lokalt kan du hoppa över det här steget.
2. Redigera eller skapa de sidor du vill ha i `content/`. Glöm inte att använda korrekt HTML5 och att deklarera PHP-variabeln `$title`.
3. Öppna `source/skel.php`, och leta dig fram till sektionen som definierar sidmenyn. Om du har skapat eller redigerat sidor som du vill ska synas i menyn, måste du uppdatera den manuellt.
4. Öppna en terminal, och navigera till mappen dit du klonade hemsidan.
5. Öppna `source/creator.php` och kontrollera att variablerna `$Target` och `$URL` har värden som är rätt för ditt system.
6. Kör `php source/creator.php` i terminalen. Du borde få en utskrift av alla filer som den har behandlat och kopierat.

## 2.3 PHP-variabler du kan använda

`$title` är webbsidans titel. Den kan med fördel användas även som rubrik, till exempel genom att skriva `<h1><?php echo $title; ?></h1>`

`$Target` är den mapp som webbsidan genereras till. Även om denna variabel inte förefaller så användbar, så är den tillgänglig för indexfilen. Troligen bör du använda `$URL` istället i de flesta fall.

`$URL` är webbsidans URL-adress. Denna är mycket användbar när du ska länka. Istället för att skriva `<a href='http://vansterliberalerna.se/sida/index.php'>...</a>`, så kan man skriva `<a href='<?php echo $URL; ?>sida/index.php'>...</a>`. Det kanske inte ser ut som någon förenkling, men det för med sig en

---

sak som är en enorm fördel: när webbsidan flyttas, uppdateras URL-er automatiskt. Alla som någon gång har flyttat en webbsida med vanliga hårdlänkar vet att det är ett helvete att byta ut dem manuellt.

## 2.4 Att använda PHP dynamiskt på servern

Som tidigare nämnts, är det fullt möjligt att använda PHP både för att kompilera webbsidan, och för att generera dynamiskt innehåll. Det lättaste är att helt enkelt använda olika öppningstaggar för PHP beroende på om det som ska genereras är dynamiskt innehåll eller statiskt innehåll.

Som standard, använder PHP `<?php ... ?>` för att markera ett block med PHP-text. Men det finns två andra varianter:

`<? ... ?>` är kortformen av standardtaggen.

`<% ... %>` är den tagg som används av Microsoft ASP, och är tillgänglig även i PHP.

PHP avråder från att man använder `<? ... ?>`, eftersom det kan skapa problem med XML om man använder det. Vänsterliberalernas webbsida gör inte det, men vi föreslår ändå att `<?php ... ?>` används för statiskt innehåll (developer-side), `<% ... %>` för dynamiskt innehåll (server-side), och `<? ... ?>` inte alls.