

Blogg

Joakim Andersson joaand-5

2015-05-28

Innehåll

1. Uppdrag.....	3
2. Mål och målgrupper	4
3. Innehåll och funktioner	5
4. Struktur och navigering	6
5. Grafisk form.....	7
5.1 flow.php.....	7
5.2 post.php.....	7
5.3 my_page.php.....	8
5.4 dynamic_user_page.php	8
6. Utformning av webbplats.....	9
6.1 Klass diagram.....	9
6.2 Data diagram	9
7. Problem och reflektioner	10
8. Sammanfattning	11
9. Appendix.....	12

1. Uppdrag

Uppdraget är att bygga en blogg med tillhörande cms backend. Webbplatsen ska vara uppbyggd med html och css, följa pul och övriga lagar som gäller webbplatser. Koden ska vara validerad på minst tre av de vanligaste webbläsarna.

Kraven för bloggen har delats upp i två sektioner, frontend och backend. Frontend visas för de användare som inte har loggat in.

Frontend

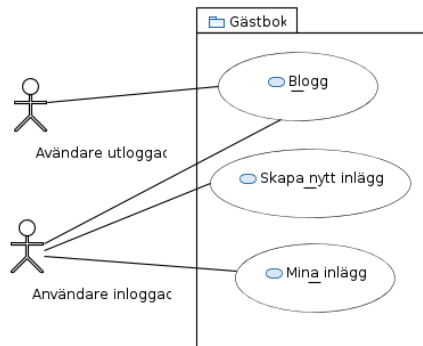
- En användare ska kunna se alla registrerade bloggare och genom att trycka på tillhörande länk ska det gå att besöka bloggarens egna sida där dennes inlägg visas.
- De senaste nyheterna ska visas.
- De senaste inläggen ska visas.
- Det ska gå att logga in och finnas möjlighet att registrera sig.

Backend

- En användare ska kunna skapa inlägg.
- Visa egna inlägg och kunna redigera och ta bort.
- Ladda upp fil eller bild.
- Visa bilder och kunna ta bort egna bilder.
- Användaren ska kunna logga ut.

2. Mål och målgrupper

Andvändningsfall



Aktör
En användare kan vara inloggad eller utloggad.
Användarens tillgång till systemet är olika om man är inloggad eller utloggad.

Mål ett.
En användare ska kunna läsa blogginlägg som andra användare skrivit.

Funktioner.
1. Visa blogg
2. Visa användare

Mål två.
En användare ska kunna skriva egna inlägg i gästboken.

Funktioner.
1. Skriv inlägg

Mål tre.
Egna inlägg ska kunna raderas.

Funktioner.
1. Visa mina inlägg
2. Radera mina inlägg

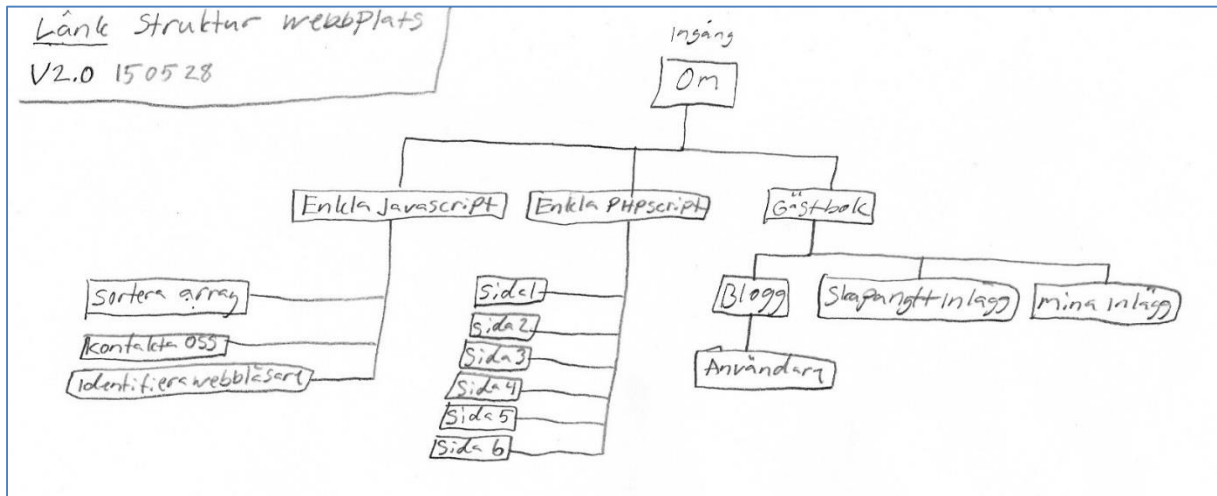
3. Innehåll och funktioner

- Visa blogg
- Visa användare
- Skriv inlägg
- Visa mina inlägg
- Radera mina inlägg

Scenario:

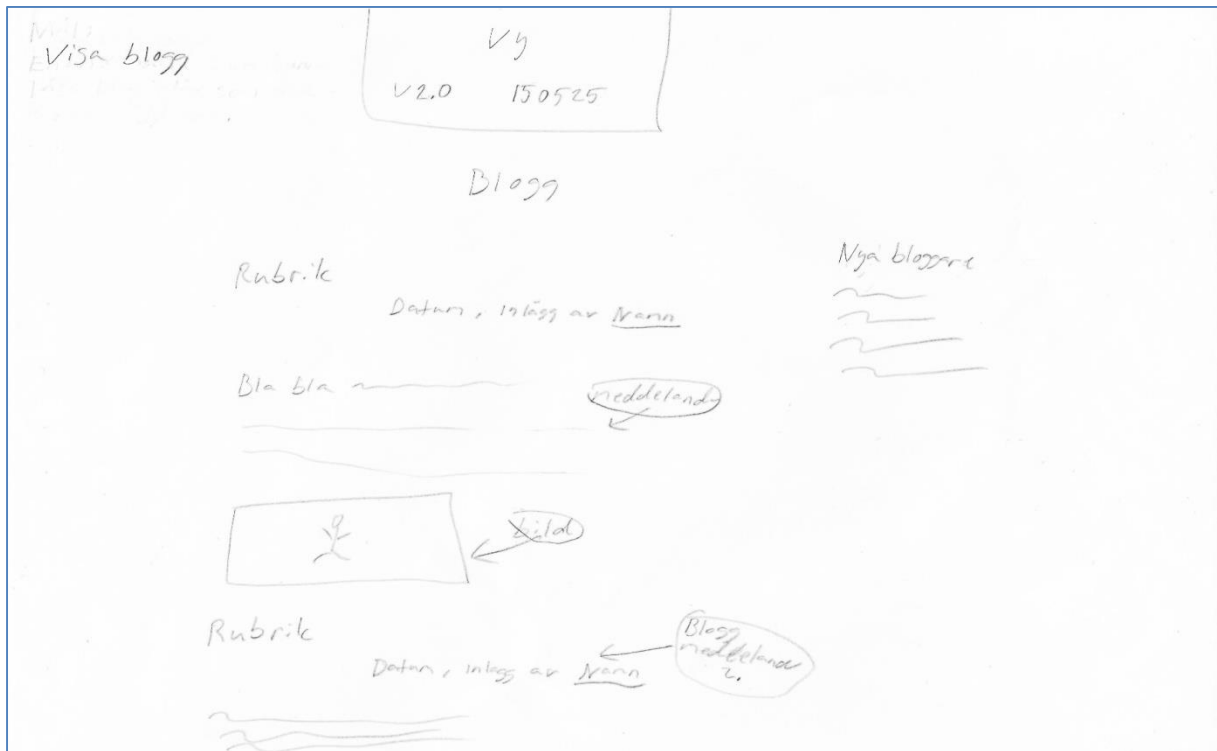
<i>Användaren Hugo</i>	<i>Systemet</i>	<i>Funktioner</i>
Mål 1. Hugo beslutar sig för att besöka en students webbsida på LTU's webserver. Där finns en gästbok och Hugo trycker på blogg i menyn för att visa vad andra användare skrivit i bloggen.	Webbservern skickar en webbsida till klienten.	Visa blogg
Mål 1. Hugo tycker att Linas blogginlägg är intressant och vill se om hon har fler inlägg. Hugo trycker på länken för att komma till Linas webb-blogg.	Webbservern hämtar alla Linas blogginlägg och publika information. Webbservern skickar därefter en webbsida till klienten.	Visa användare
Mål 2. Hugo vill skriva ett eget inlägg och loggar därför först in i backend. Hugo trycker sedan i gästbokens meny, skapa nytt inlägg. Hugo skriver in sitt inlägg i formuläret och trycker slutligen på knappen inlägg.	Att logga in ligger utanför bloggens systemgräns. Webbservern skickar en webbsida med ett formulär till klienten. Klienten skickar formuläret till servern och inlägget sparas i databasen.	Skriv inlägg
Mål 3. Hugo trycker på Mina inlägg i menyn.	Webbservern skickar en webbsida till klienten.	Visa mina inlägg
Mål 3. Hugo befinner sig ännu på sidan mina inlägg. Han granskar sitt senaste inlägg och ser ett stavfel. Hugo bestämmer sig för att radera sitt senaste inlägg. Han trycker på den röda knappen Ta bort, som finns strax ovanför inlägget.	Klienten skickar information om vilket inlägg som ska tas bort. Servern tar bort inlägget och skickar en webbsida till klienten.	Radera mina inlägg
Avsluta: Hugo loggar ut från webbsidan och stänger sedan webbläsaren.	Att logga ut och stänga webbläsaren ligger utanför bloggens systemgräns.	-

4. Struktur och navigering

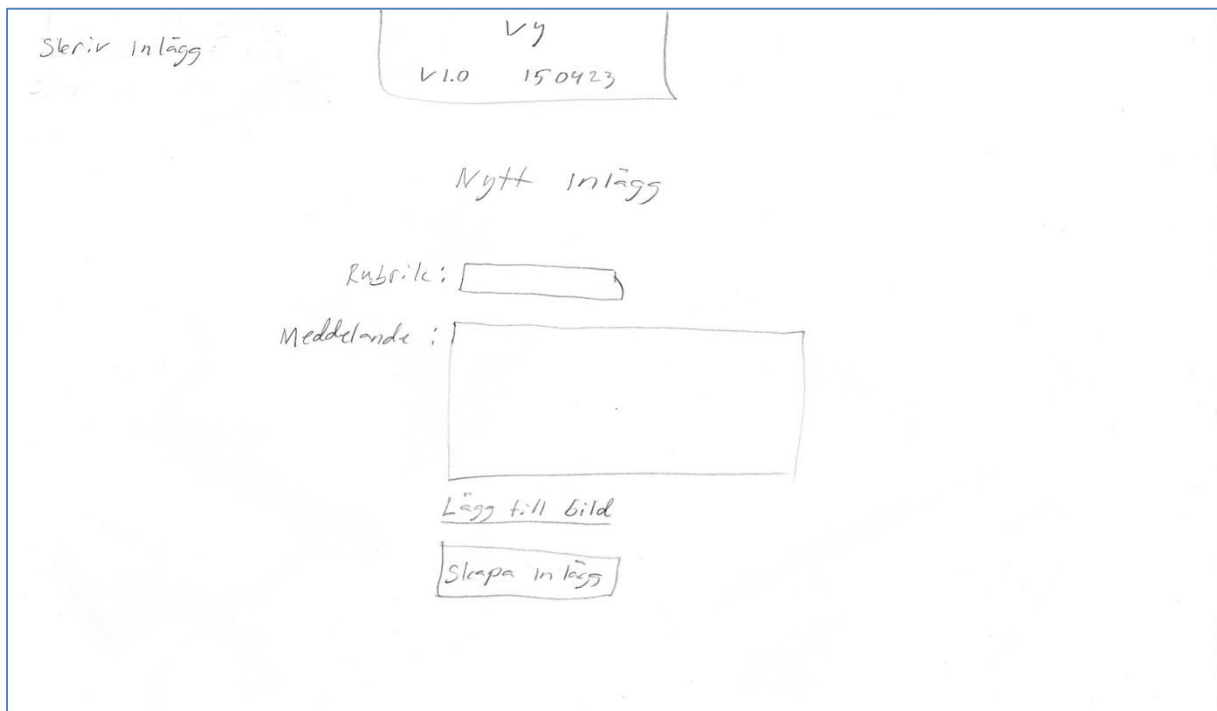


5. Grafisk form

5.1 flow.php



5.2 post.php



5.3 my_page.php

Radura mina inlägg



Vy


V1.0 150423

Namn

Rubrik

Datum, ta bort inlägg ☐









Ändring
inlägg

Rubrik

Datum ta bort inlägg ☐





5.4 dynamic_user_page.php

Visa användare

Vy

V1.0 150525

Namn

Rubrik

Datum







flera inlägg

Rubrik

Namn

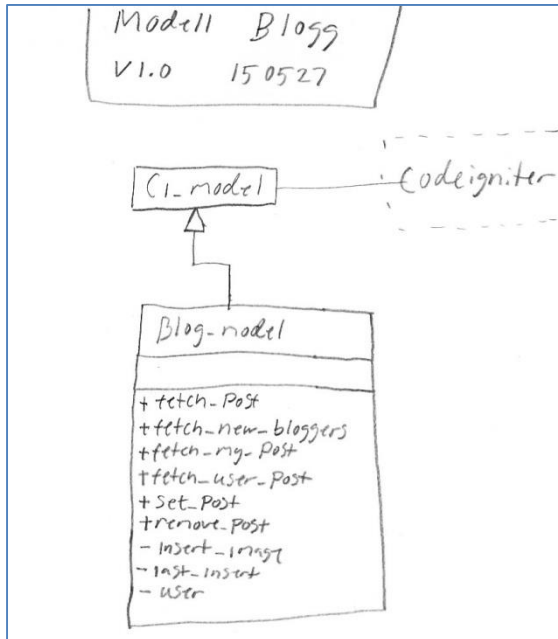
Datum



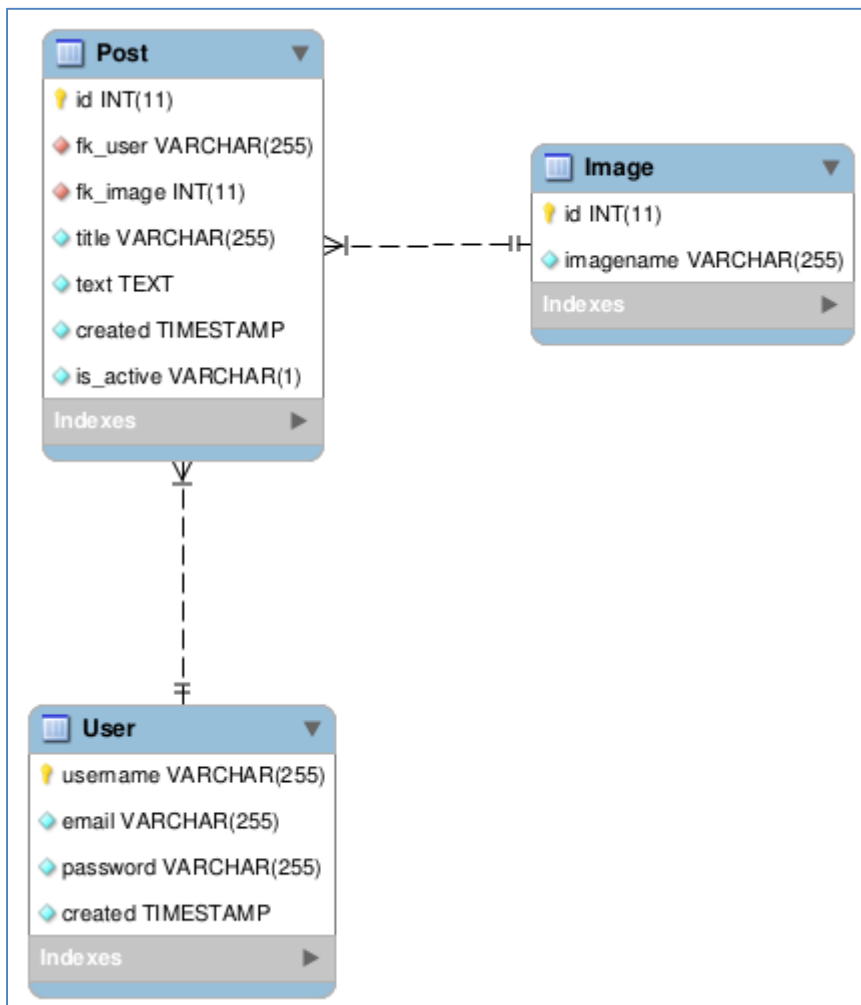


6. Utformning av webbplats

6.1 Klass diagram

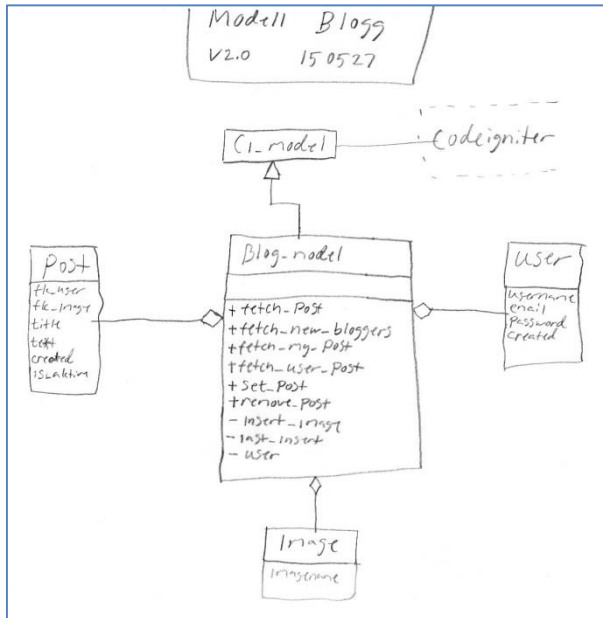


6.2 Data diagram



7. Problem och reflektioner

- Inser att det här med dokumentation är viktigt, det ger struktur och ordning.
- Jag tror man ska sträva efter att ha samma namn på klasserna i modellen som i entiteterna i datamodellen. Har missat det och inser att jag borde ha börjat med uppgift2 istället och sen ha kört den samtidigt med att man kodat i uppgift1. Klass modellen kunde ha sett ut så här istället, för att mer koppla samman datamodellen med klass modellen.



8. Sammanfattning

Uppdraget

Uppdraget var att bygga en blogg med tillhörande cms backend. Webbplatsen skulle byggas upp med html och css, följa pul och övriga lagar som gäller webbplatser. Koden ska valideras på minst tre av de vanligaste webbläsarna.

Kvav

På grund av tidsbrist har inte alla kraven kunnat implementeras i koden.

Enligt Post- och telestyrelsen är webbplatsinnehavare skyldiga att redovisa [kakor](#). Jag har inte med att kontrollera vilka kakor som används av Codeigniter eller inloggningssystemet, därför finns inget system för att redovisa kakor för användarna.

Ett annat av kraven för uppdraget var att en användare ska kunna se alla registrerade bloggare. Istället för att visa alla bloggare på webbsidan har bloggarna begränsats till att visas endast de senaste 7 bloggarna.

Slutord

Avslutningsvis vill jag säga att uppgiften varit kul och utvecklande.

9. Appendix

