

Eksamen, DSDS, forår 2007

Scripting, Databaser og Systemarkitektur

af Martin Elsman og Arne Glenstrup

- Alle hjælpemidler tilladt, dog ikke computer og kommunikationsmidler
- Eksamen er skriftlig, fire timer og bedømmes fra 0 til 100 procent
- Eksamen består af fem opgaver der alle ønskes løst
- Indenfor hver opgave vægtes delopgaverne lige

- Et godt råd: Gennemlæs opgavesættet inden du begynder at besvare de enkelte opgaver

Introduktion

Baggrund: I dette eksamenssæt skal du implementere dele af et webbaseret afstemningssystem kaldet VoteAboutIt.com. Ideen med systemet er at brugere kan oprette afstemninger som kan besvares af andre brugere af systemet. Når en bruger af systemet har oprettet en afstemning i form af et spørgsmål og en række modtagere af spørgsmålet, sendes der email til modtagerne, som derved henvises til afstemningssitet.

Opgave 1 (15 procent) - HTML

Baggrund: Nedenfor vises en aktuel hovedside for det web-baserede afstemningssystem:



Hovedsiden viser en brugers indbakke (Inbox) og udbakke (Outbox) og giver mulighed for at en bruger kan oprette en ny afstemning ved at indtaste et spørgsmål i indtastningsfeltet og trykke på knappen "Create Question". Bemærk at systemet kun giver mulighed for at stille spørgsmål hvorpå der kan svares ja eller nej.

Hovedsiden er implementeret ved filen `index.php`, som tager to formvariabler:

1. Et unikt tal til identifikation af en person (`pid`)
2. Et password til autentifikation af en person (`passwd`)

Disse to formvariabler sendes rundt i mange af systemets `php`-filer og benyttes til identifikation og autentifikation. Vi ser i det følgende bort fra systemets egentlige login-side hvor en bruger anmodes om sin email-adresse og sit password.

Opgave 1.1

Din opgave: Opskriv HTML koden for formen i siden ovenfor. Besvarelsen skal udelukkende indeholde HTML-koden for formen og ikke HTML-koden for hele siden:

```
<form action="votes.php">
</form>
```

Formen skal indeholde de to skjulte formvariabler `pid="3"` og `passwd="9347"` samt følgende to formelementer:

1. Et indtastningsfelt af type `text` og attribut `name="head"`. Feltet skal være 50 tegn bredt
2. En submit-knap med attribut `value="Create Question"`

Aktion (`action`) for formen skal være filen `votes.php`.

Opgave 1.2

Baggrund: Indbakken består af en tabel med tre kolonner. Den første kolonne angiver de forskellige indkomne spørgsmål og den anden kolonne indeholder `mailto`-links for de enkelte spørgsmålstillere. Den tredje kolonne indeholder links for forskellige brugeraktioner tilknyttet de forskellige spørgsmål.

Din opgave: Opskriv HTML-koden for indholdet af indbakken (Inbox) i billedet ovenfor.

Der stilles yderligere følgende krav til HTML-koden:

- De to `mailto`-links for personerne Niels Hallenberg og Arne Glenstrup skal angives med henholdsvis `nh@itu.dk` og `panic@itu.dk` som email-adresser.
- De to status-links skal henvise til filen `status.php`, som udover de to formvariabler `pid` og `passwd`, der videresendes fra `index.php`-siden, tager en formvariabel `qid` (question id) som argument. Spørgsmålet "Did you get any X-Mas presents this year?" har tilknyttet `qid="2"` og spørgsmålet "Have you heard about the course Distributed Systems?" har tilknyttet `qid="3"`.
- De to links YES og NO skal henvise til filen `answer.php`, der udover de tre formvariabler som filen `status.php` tager som argument, også tager en formvariabel `ans` som argument. Formvariablen `ans` antager værdien `y` for YES og `n` for NO.

Opgave 2 (15 procent) - PHP

Opgave 2.1

Baggrund: Til brug for `VoteAboutIt.com` ønskes der implementeret en funktion `gen_passwd` til generering af passwords til brugere der oprettes på sitet.

Din opgave: Skriv funktionen `gen_passwd`. Den skal ikke tage nogle argumenter, og skal returnere et password bestående af fire tilfældigt valgte cifre. Undgå dog at

generere passwords indeholdende cifrene 0 og 1; disse cifre kan forveksles med bogstaverne o, O, og l. Funktionen skal benytte en while-løkke.

Bemærk: Du kan passende benytte funktionskaldet $\text{rand}(N, M)$, hvor M og N er heltal, til at generere et tilfældigt ciffer mellem M og N (begge inklusive). Eksempelvis returnerer kaldet $\text{rand}(5, 8)$ et tal mellem 5 og 8 (begge inklusive).

Opgave 2.2

Din opgave: Givet PHP-funktionerne `mysql_query` og `mysql_fetch_row`, skriv en funktion `mydb_one_field`, som tager en SQL select-kommando som argument og returnerer det første felt i den første række i resultatsættet. Funktionskoden kan antage at en forbindelse allerede er etableret til databasen. I tilfælde af fejl ved kald til funktionerne `mysql_query` og `mysql_fetch_row` skal funktionen `error` kaldes med en beskrivende fejlbesked.

Bemærk: Du kan antage at funktionen `error` tager en fejlbesked (en streng) som argument og returnerer en passende fejlbesked til brugerens browser hvorefter funktionen afslutter scriptudførelsen ved et kald til PHP-funktionen `exit`.

Opgave 3 (10 procent) - Regulære udtryk

Opgave 3.1

Baggrund: Det regulære udtryk `[0-9]` matches af en forekomst af et af cifrene 0-9.

Din opgave: Opskriv et regulært udtryk som matches af genererede passwords bestående af 4 cifre.

De to passwords 9347 og 2856 skal således matche det regulære udtryk, og 3019 må også gerne matche det regulære udtryk. Derimod skal passwords bestående af andre tegn end tal-cifre og passwords der består af et antal cifre forskelligt fra 4 ikke matche det regulære udtryk.

Opgave 3.2

Din opgave: Skriv en PHP-funktion `va_chk_passwd` til at undersøge om et password har formen som beskrevet i Opgave 3.1. Hvis passwordet, som overføres til funktionen som argument, har den rigtige form, skal funktionen med det samme returnere til stedet hvorfra funktionen kaldes. Ellers skal funktionen sende en fejlbesked til brugerens web-browser ved brug af den indbyggede PHP-funktion `echo`, hvorefter programmet skal stoppe ved et kald til PHP-funktionen `exit`.

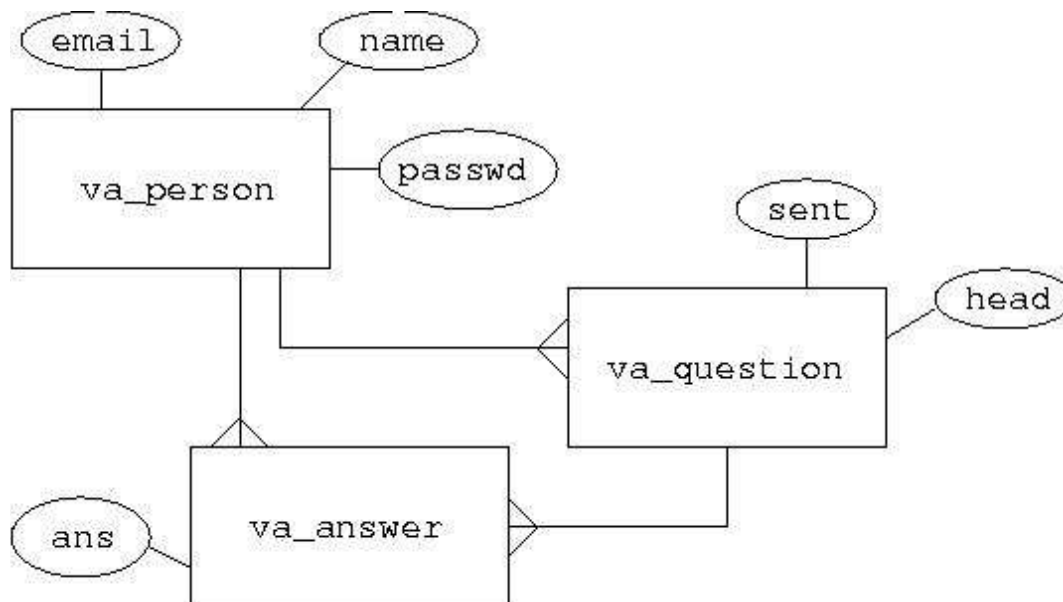
Til at undersøge om passwordet har den rigtige form skal du benytte PHP-funktionen `ereg`.

Opgave 4 (25 procent) - SQL

Baggrund: Datamodellen for afstemningssystemet består af tre tabeller:

1. `va_person` - indeholder data om oprettede brugere
2. `va_question` - indeholder data om spørgsmål stillet af brugerne
3. `va_answer` - indeholder information om hvilke brugere der ønskes at svare på de forskellige spørgsmål, samt eventuelle svar.

For at simplificere systemet antages det i det følgende at der kun kan stilles spørgsmål hvortil der kan svares ja eller nej. Datamodellen er angivet nedenfor i form af et E/R-diagram. De firkantede kasser angiver entiteterne (tabellerne) og de ovale cirkler angiver attributter (felter) tilknyttet de enkelte tabeller. Herudover er en-til-mange relationer angivet med kragefødder.



Tabellerne `va_person` og `va_question` oprettes i MySQL med følgende SQL kommandoer:

```

create table va_person (
  pid integer auto_incremant primary key,
  name varchar(200) not null,
  email varchar(200) unique not null,
  passwd varchar(10) not null
);
create table va_question (
  qid integer auto_incremant primary key,
  pid integer not null,
  sent date,
  head varchar(200) not null
);
  
```

I ovenstående SQL-kommandoer benyttes MySQL's indbyggede `auto_incremant` feature til at skabe nye unikke identifikationsnumre for hver række i tabellerne.

Følgende SQL insert kommandoer kan efterfølgende antages at være udført:

```

insert into va_person (pid, email, name, passwd)
values (1, 'panic@itu.dk', 'Arne Glenstrup', '4567');

insert into va_person (pid, email, name, passwd)
values (2, 'nh@itu.dk', 'Niels Hallenberg', '2345');

insert into va_person (pid, email, name, passwd)
values (3, 'mael@itu.dk', 'Martin Elsman', '9347');

insert into va_question (qid,pid,head)
values (1, 3, 'Do you like PHP?');

insert into va_question (qid,pid,head)
values (2, 2, 'Did you get any X-Mas presents this year?');

insert into va_question (qid,pid,head)
values (3, 1, 'Have you heard about the course Distributed Systems?');
  
```

Opgave 4.1

Din opgave: Vis resultatet af at udføre følgende select kommando:

```
select name, email from va_person order by name;
```

Opgave 4.2

Din opgave: Opskriv en SQL kommando til at oprette tabellen `va_answer`. Tabellen skal have tre kolonner (se iøvrigt E/R-diagrammet ovenfor):

- `qid` - henvisning til `va_question` tabel; benyt datatypen `integer` til dette felt
- `pid` - henvisning til `va_person` tabel; benyt datatypen `integer` til dette felt
- `ans` - svar på spørgsmål; `ans` skal kunne antage værdierne 'n' for NO, 'y' for YES og NULL for at indikere at der endnu ikke er svaret på spørgsmålet; benyt datatypen `char(1)` til dette felt

Benyt endvidere MySQL's `unique-constraint` til at sikre at der ikke forekommer to rækker i tabellen `va_answer` med samme kombination af `pid`- og `qid`-numre.

Opgave 4.3

Baggrund: I denne opgave skal du opskrive SQL `insert`-kommandoer til tilknytning af svarpersoner til allerede oprettede spørgsmål. Det kan antages at personerne ved indsættelsen endnu ikke har svaret på spørgsmålene.

Din opgave: Opskriv SQL `insert`-kommandoer til følgende tilknytninger:

1. Personerne Niels Hallenberg og Arne Glenstrup skal svare på spørgsmålet "Do you like PHP?"
2. Personerne Martin Elsmann og Arne Glenstrup skal svare på spørgsmålet "Did you get any X-Mas presents this year?"

Opgave 4.4

Din opgave: Konstruer en SQL `select`-kommando til udtrækning af indbakke-rækker for personen Martin Elsmann. Følgende kolonner skal indgå i resultatet: `qid`, `head`, `ans`, `name` og `email`. Kolonnerne `name` og `email` skal indeholde navn og email-adresse for personen der har stillet spørgsmålet.

Bemærk: Det er nødvendigt med et `join` mellem alle tre tabeller.

Opgave 4.5

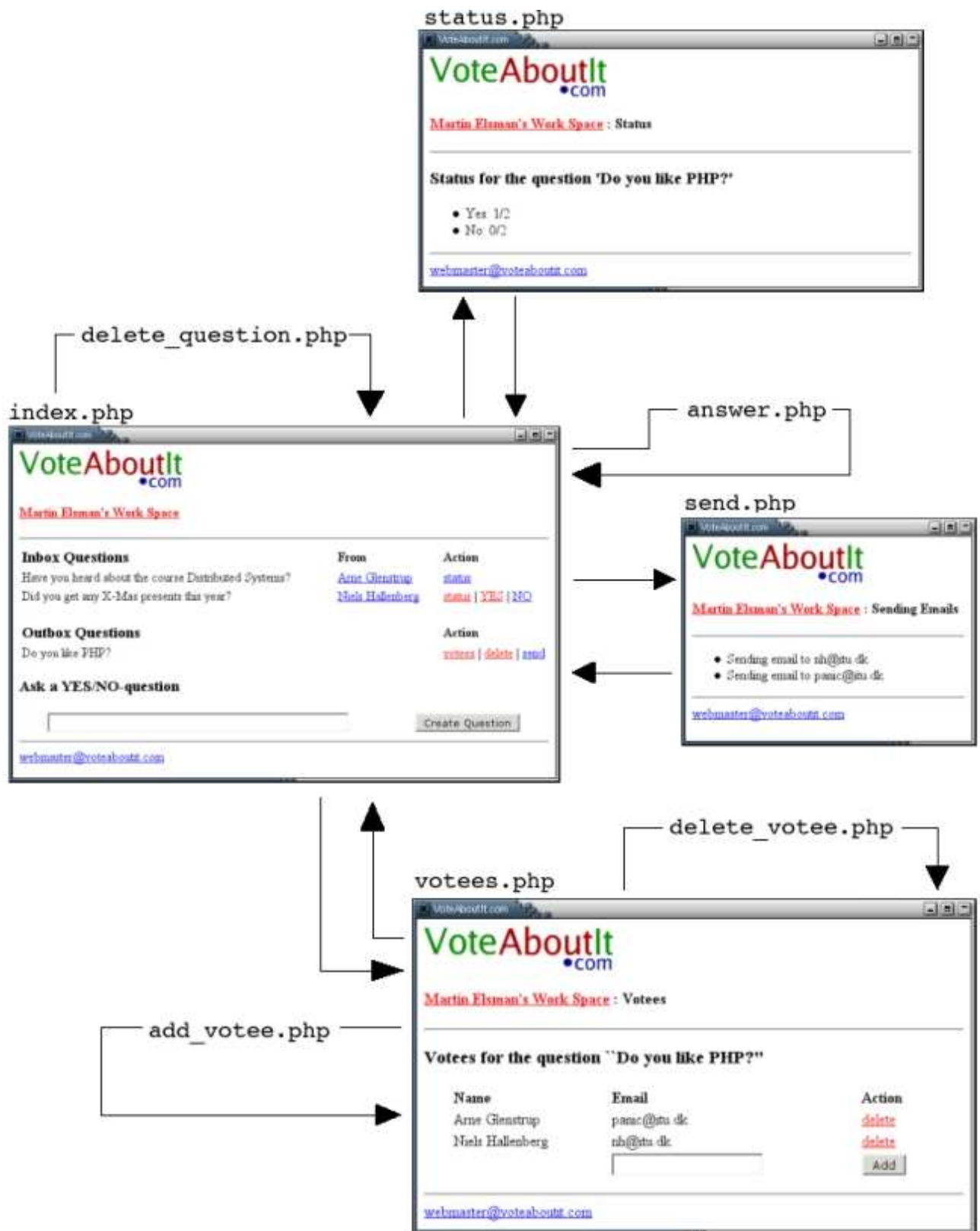
Baggrund: Til brug for visning af antal ja-svar på spørgsmålet "Do you like PHP?" kan denne SQL-kommando bruges:

```
select count(*)
  from va_answer
 where ans = 'y' and qid = 1;
```

Din opgave: Opskriv to SQL `select`-kommandoer til beregning af henholdsvis (1) antal nej-svar og (2) det samlede antal personer der har mulighed for at svare på spørgsmålet.

Opgave 5 (35 procent) - Web-service

Baggrund: Til konstruktion af PHP-filerne som implementerer afstemningssystemet er der opstillet et tilstandsdiagram til at underbygge de forskellige systemfunktioner. Kasserne i tilstandsdiagrammerne repræsenterer web-sider (implementeret som PHP-filer), pilene uden annoteringer repræsenterer links og de annoterede pile repræsenterer transaktioner. Tilstandsdiagrammet er som følger:



For at begrænse opgaven ser vi i det følgende bort fra den nederste del af systemet der omhandler tilknytningen af personer til et spørgsmål.

Det kan antages at biblioteket `va_lib.php` inkluderer `mydb.php`-biblioteket fra forelæsningsne samt funktionerne `gen_passwd` (fra opgave 2.1) og `mydb_one_field` (fra opgave 2.2).

Opgave 5.1

Baggrund: Nedenfor følger en skabelon til en funktion `va_header`, som har til formål at returnere en header (begyndende HTML-kode) med VoteAboutIt-logo og

overskrift til en brugers browser. Efter en PHP-fil har kaldt funktionen `va_header` kan filen benytte den indbyggede PHP-funktion `echo` til at returnere yderligere HTML-kode til brugerens browser.

Funktionen `va_header` tager tre argumenter: et personidentifikationsnummer `$pid`, brugerens password `$passwd` og en heading `$head` med overskriftinformation. Funktionen etablerer en databaseforbindelse til brug for at autentificere en bruger. Herefter sendes begyndende HTML-kode til visning af logo og overskrift.

Her er en skabelon til funktionen `va_header`:

```
function va_header ($pid, $passwd, $head) {
    va_chk_int($pid);
    va_chk_passwd($passwd);
    va_chk_text($head);
    mydb_connect();

    // Authenticate user and find user's name
    $query = " ... ";

    $rows = mysql_query($query);
    if ( $rows ) {
        $row = mysql_fetch_row($rows);
        $name = $row[0];
        echo "<html><body bgcolor='white'><img src='vai200.png'>
            <h4><a href='index.php?pid=$pid&passwd=$passwd'>$name's
                Work Space</a>";
        if ( $head != "" ) {
            echo " : $head";
        }
        echo "</h4><hr>";
    } else {
        error ("Couldn't authenticate user");
    }
}
```

Det kan antages at funktionerne `va_chk_int`, `va_chk_passwd` og `va_chk_text` er til rådighed og at de returnerer en fejlside til brugerens browser i tilfælde af at argumenterne ikke har den forventede form.

Din opgave: Opskriv PHP-koden for definitionen af PHP-variablen `$query` i funktionen `va_header`, således at såfremt funktionens argument `$passwd` stemmer overens med brugerens password i databasen, da vil brugerens navn være tilgængeligt i PHP-variablen `$name` efter udførelsen af PHP-sætningen "`$name = $row[0];`" i funktionen.

Opgave 5.2

Din opgave: Konstruer en funktion `va_footer` til at udskrive afsluttende HTML-kode til en browser, inklusiv

1. en horisontal linie (tag `<hr>`)
2. et `mailto`-link til email-adressen `webmaster@voteaboutit.com`

Bemærk: Funktionen skal benytte PHP-funktionen `echo` til at udskrive HTML-kode til browseren.

Opgave 5.3

Baggrund: I denne opgave skal du implementere dele af den fil der afsender emails til de personer som skal svare på en brugers spørgsmål. Følgende funktion `email_body` tager et personidentifikationsnummer `$pid` og et password `$passwd`

som argumenter og returnerer en streng til brug som krop i en email:

```
function email_body ($pid, $passwd) {
    return "  Hi there,

    You have been invited to answer a question at
    VoteAboutIt.com. To answer the question, access

        http://www.voteaboutit.com/index.php?pid=$pid&passwd=$passwd

    Best Regards,

    A friend at VoteAboutIt.com";
}
```

Filen `send.php`, som skal sørge for afsendingen af emails, tager tre formvariabler: et personidentifikationsnummer `pid`, brugerens password `passwd` og et spørgsmålsidentifikationsnummer `qid`.

En skabelon til filen kan se således ud:

```
<?
include("va_lib.php");
va_chk_int($qid);
va_header($pid, $passwd, "Sending Emails");

$rows = mysql_query(" ##A## ");

echo "<ul>";
while ( $row = mysql_fetch_row($rows) ) {
    ##B##
}
echo "</ul>";

// Update question status
mysql_query("update va_question set sent = now()
            where qid = $qid
            and pid = $pid");
va_footer();
?>
```

Din opgave: Skriv kode for programpunkterne `##A##` og `##B##`.

Bemærk: Til brug for programpunkt `##B##` skal du benytte PHP-funktionen `mail` til afsending af emails. Ydermere skal der for hver afsendt email udskrives en linie til brugerens browser med indikation af afsendelsen. Funktionen `mail` tager tre argumenter: en email-adresse for modtageren, en titel og en krop.

Opgave 5.4

Din opgave: Konstruer filen `status.php`. Filen tager tre formvariabler: et personidentifikationsnummer `pid`, brugerens password `passwd` og et spørgsmålsidentifikationsnummer `qid`.

Filen skal vise antallet af brugere der har svaret ja på et spørgsmål, hvormange der har svaret nej og antallet af mulige svar.

Bemærk: Det er vigtigt at det sikres at en bruger af systemet ikke kan se status for andre brugeres spørgsmål.

mael@itu.dk, panic@itu.dk