Inloggning

## Utförande

Först gjorde jag planen över hur vill jag att sidan ska fungera. Jag har först följt bokens rekommendation och skapat fil med inloggningsuppgifter för att sedan endast behöva skriva en linje av kod för att koppla mig till databasen.

Efter det skapade jag index filen som laddas upp först när man startar servern och där dyker upp det två input rader med username och lösenord samt knappen att skicka informationen.

### Index

Först har jag skapat kopplingen till databasen med alla uppgifter från login.php och kollat med die funktionen om kopplingen fungerar och eventuella meddela om problemet.

Med hjälp av isset funktionen och $POST har jag tagit fram användarens inskickade uppgifter och sparat det som separata variabeln och direkt använt sanitize funktionen som jag själv har skapat med hjälp av stripslashes, htmlentities, strip\_tags och real\_escape\_string. Detta förhindrar alla eventuella försök till manipulering från användaren.

När input har blivit säker och fri att använda kollar query om användarens namn finns i databasen. Med hjälp av IF meddelas användaren om till exempel användarnamn och lösenord inte är korrekta eller får informationen att till exempel något fick fel. När vi har hittat användarens namn i databasen sparar vi informationen i en row array och avslutar kontakt med databasen. Med hjälp av sparade salt och hash funktionen sparar vi lösenordet i ny form som nu jämför dator med information sparat i databasen. Om uppgifter inte stämmer får användaren endast information om att något av dessa två är fel, utan att ange att det är endast lösenord som inte stämmer för att inte ge för mycket information.

När lösenordet stämmer skapas session med användarens namn och lösenordet och sedan skickas användaren direkt till ny sida – access.php

### Access

På denna sida skapas session direkt och information sparas så att anvädaren kan nu fräsha upp sidan utan att det loggas ur. Med if funktionen kontrolleras det om Session finns och om den inte är så – skickas användaren direkt till inloggningssida. Om session finns – då visas information till användaren om dess uppgifter.

I början hade jag tanke att avsluta programmet här, men det tyvärr kunde man inte själv bestämma när session skulle gå ut, utan starta om allt. Därför tänkte jag att det skulle vara användarbar att skapa en knapp där man kan själv logga ut och radera sessionen.

Det skapades med enkel a och hänvisning till nästa sida – logout.php

### Logout

Logout är en enkel sida med endast php som raderar alla sessioner, cookies och loggar ut användaren och sedan på direkten skickar den till index sida utan att visa något.

### Create users

Denna sida skapades endast för att skapa användaren till databasen. Den körs inte en enda gång i hela programmet utan jag använde den en gång för att få inloggningsuppgifter i basen.

Den behövs alltså inte att startas för att programmet ska fungera!

Programmet skapar först kontakt med sql bas där den skickar inloggningsuppgifter och eventuell meddelar om problemet om det inte går.

Efter det skapas en query med att skapa en ny table med 3 styck rader – id, username, password och sedan skickas det till databasen och kontrolleras om allt går bra.

Nästa stege är att skapa användaren – d.v.s. ange användarens namn och lösenord. Efter det anger programmet salt1 och salt2 för att tillsammans med hash – göra lösenord krypterade.

Till slut med hjälp av funktion new\_user – slå ihop dessa informationer till en inloggningsuppgift i databas och spara det där.

## Säkerhet

Jag har lagt stor fokus på att öka säkerheten i programmet. Från att skapa en stark funktion som ”rensar” både usernamn och lösenord från användaren från alla försök till att manipulera programmet med hjälp av flera funktioner.

Jag har också säkerställt att sektionen går att avluta i förtid med logga ut knappen som jag har testat på flera webbläsaren och flera gånger.

Tack vore funktioner som striptags eller htmlentitles förhindras förök till HTML injections och eal\_escape\_string eller stripslashes hjälper också med att förhindra SQL injections.

Jag har också använd både salt – i början och slutet av lösenord men också hash funktionen som gör det mycket svårare att kunna få tillgång till riktiga lösenord.

## Utveckling

Som nämnt förut jag har varit nöjd med endast att få användaren komma in på inlogg sida och få det sparat i session, men efter liten stund tänkte jag att det skulle vara säkrare att kunna logga ut men även var det irriterande att om jag har loggat ut och tryckt knappen tillbaka, så fortfarande var jag utloggat trots allt. Därför tänkte jag att göra extra knappen att logga ut för att öka säkerheten.

Utöver detta såklart finns massor utrymme för att utseendemässigt ändra massor på alla sidor, jag har dock fokuserat på det tekniska bakom detta och sett till att allt fungerar som det ska.