

Anonüümsete sõnumite valideerimine

Koduseks tööks jäi ühe osana koostada funktsioon, mis muudab valitud sõnumi kirjet andmetabelis nii, et paneb kirja, kas sõnumit on lubatud näidata või mitte, kes valideeris ja millal.

Et oleks selgem, mis toimub, siis anname ülevaate töövoost.

Kõigepealt tuleb oma veebilehele sisse logida, sest just lisatud sisselogimist kontrollivad osad lehtede alguses ei lasegi enne valideerimise lehte vaadata, kui on sisse logitud (on määratud sessiooni muutujad).

Sisselogimise järel satute main.php lehele (meie pealehele.

Teiseks tuleb main.php lehel klõpsata lingil, mis viib sõnumite valideerimisele!

Link on lisatud sellise html koodiga:

```
Valideeri anonüümseid <a href="validatemsg.php">sõnumeid</a>  
See koodijupp ise on meil ilmselt loetelu <ul> ühe loendielemendi <li> sees.
```

Nüüd satute lehele validatemsg.php, millele kuvatakse loend kõigist valideerimata sõnumitest ja seda tagurpidi järjekorras – uuemad ehk suurema id-väärtusega enne.

Sõnumid

Siin on minu [TLÜ](#) õppetöö raames valminud veebilehed. Need ei oma mingit

- Igaüks peab ise ennst kiitma, muidu jääb see tähtis asi tegemata!
[Valideeri](#)
- Tere tere tere
[Valideeri](#)
- tervist
[Valideeri](#)
- Töörõõm on suur aga võidurõõm on suurem!
[Valideeri](#)
- Ega see veeb nüüd nii keeruline ka pole - päris lihtsaid asju näeb siin.
[Valideeri](#)
- Alanud on uus nädal.
[Valideeri](#)
- Alanud on uus nädal ja käes on muusikapäev!
[Valideeri](#)

Funktsioon, mis seda teeb, on `readallunvalidatedmessages()`. SQL käsk, mis sellised sõnumid andmetabelist küsib, näeb välja selline:

SELECT id, message FROM vpmsg3 WHERE valid IS NULL ORDER BY id DESC

Siin palun jälgige:

- Andmetabelist küsitakse id ja sõnum ise.
- Andmetabel on näites „vpmsg3“, siin peab olema see nimi, mis Teie enda andmebaasis samasugusel tabelil on (enamusel ilmselt sama nimi ilma numbrita).
- Otsitakse vaid neid, millel validate väljal on tühi väärtus (NULL).
- Järjekord id väärtuste järgi tagurpidi (ORDER BY id DESC)

Sõnum ise pannakse nähtavale ühe loetelu punktina, tema järele lisatakse link(element <a>), mis kutsub välja sõnumi lubamise/keelamise lehe (validatemessage.php) selliselt, et lisab URL-ile (ehk saadab GET meetodil) valitud sõnumi id.

HTML kood, mis ühe sõnumi kohta kirjas, on selline:

```
<li>Igaüks peab ise ennst kiitma, muidu jääb see tähtis asi tegemata!<br><a href="validatemessage.php?id=12">Valideeri</a></li>
```

Siintoodud näide viib edasi aadressile: validatemessage.php?id=12

Kolmandaks oleme jõudnud lehele validatemessage.php, millele saadeti GET meetodiga valitud sõnumi id.

Sõnumid

Siin on minu [TLÜ](#) õppetöö raames valminud veebilehed. Need ei ole

- [Tagasi](#) sõnumite lehele!

Valideeri see sõnum:

Igatüks peab ise ennst kiitma, muidu jääb see tähtis asi tegemata!

- ☒ Keela näitamine
☐ Luba näitamine

Kinnita

Kuna uue lehe laadimisel unustab php enamuse infost (alles jäävad sessioonimuutujad ning edastatud GET-meetodil saadetud info), siis just URL-ile lisatud id väärtuse järgi loebki see leht valitud sõnumi andmebaasist uuesti sisse, ning lisab raadio nupud lubamiseks ja keelamiseks.

Funktsioon, mida kasutame, on readmsgforvalidation ja sellele saadetakse infona aadressirealt URL-ist vajalik id (\$_GET["id"])

SQL käsk, mis konkreetse sõnumi id järgi loeb, on:

```
SELECT message FROM vpamsg3 WHERE id = ?
```

Taas jälgige, et oleks õige tabeli nimi.

Neljandaks tuleb nüüd valida, kas lubada või keelata ja vajutada kinnituspuppu (selle nimi on submitValidation).

Meil on olemas vorm, milles on ka üks „kaval“ peidetud sisend, millele me oleme väärtuseks andnud just käsitletava sõnumi id (mille just saime GET kaudu):

```
<input name="id" type="hidden" value="<?php echo $_GET["id"]; ?>">
```

See id on vaja ju meeles hoida ja pärast taas edasi saata.

Kogu see osa, mis juhtub, kui nuppu on vajutatud, on puudu.

Vaja püüda kinni \$_POST info, käivitada funktsioon (millele mina ise panin nimeks validatemsg). Sellele funktsioonile tuleb edastada sõnumi id ja siis valideerimisinfo (kas lubatud – validation väärtus 1 või keelatud – validation väärtus 0).

Tuletan meelde, et see tabeli struktuur on greeny.cs.tlu.ee/phpMyAdmin lehel ilusti näha:

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 created	timestamp			No	current_timestamp()	
<input type="checkbox"/>	3 message	varchar(256)	utf8_estonian_ci		No	None	
<input type="checkbox"/>	4 validator	int(11)			Yes	NULL	
<input type="checkbox"/>	5 valid	int(1)			Yes	NULL	
<input type="checkbox"/>	6 validated	timestamp			Yes	NULL	

Funktsioon saab selle info, pöördub andmebaasi poole ja kasutades SQL-päringut, muudab selle id-ga sõnumi kirjet.

```
UPDATE vpamsg3 SET validator=?, valid=?, validated=now() WHERE id=?
```

- Siin validator on väli, millele tuleb saata kasutaja id, kes lubas või keelas. Selle saab sessioonimuutujast \$_SESSION["userId"]

- Valid on väli, kus kirjas, kas lubatud – 1 või keelatud – 0, selle saab nüüd POST infost
- Validated on väli, kus kirjas aeg, millal lubati/keelati. Sinna kirjutame SQL enda funktsiooniga väärtuseks praeguse hetke now().
- Loomulikult toimub see selle sõnumi kirjes, mille id-d me kasutame.
- Jälle jälgige, et tabeli nimi oleks õige!!!