

# Slanje datoteke na server (FILE UPLOAD)

Slanje datoteke na server je lako uz pomoć programskog jezika PHP i HTML forme. Međutim, treba biti dosta obazriv prilikom slanja datoteka na server zbog problema bezbednosti.

Proces dodavanja fajla na server sadrži više koraka:

- Korisnik otvara stranu koja sadrži HTML formu, gde može da izabere datoteku.
- Korisnik klikne dugme za biranje i bira datoteku koji šalje sa svog računara na server.
- Puna putanja do datoteke se pojavi u polju pored, zatim korisnik klikne na dugme za slanje.
- Izabrana datoteka se šalje na privremenu lokaciju na server.
- PHP kod stranice koja prihvata formu proverava da li je datoteka stigla i kopira je u namenjeni direktorijum.
- PHP kod potvrđuje korisniku da je datoteka stigla.

HTML forma koja omogućuje korisniku da izabere datoteku koji želi da doda na server prikazana je u sledećem primeru:

```
<!DOCTYPE html>
<html>
<head>

</head>
<body>
<form action="upload.php" method="POST" enctype="multipart/form-data">
  <input type="file" name="fileToUpload">
  <input type="submit" name="submit" value="Dodaj fajl">
</form>
</body>
</html>
```

Postoje dva pravila pri slanju datoteke u formi.

1. Zbog bezbednosti korisimo uvek metod POST
2. Formi je neophodan još jedan atribut: `enctype="multipart/form-data"`. On određuje koji tip sadržaja će se koristiti prilikom slanja forme. Omogućava da se podaci prilikom slanja na server kriptuju (može se koristiti samo uz metod POST).

Primititi još da se koristi atribut `type = "file"` koji polje za unos prikazuje kao polje za odabir datoteke. Forma šalje podatke datoteci koji se zove "upload.php", koja izgleda ovako:

```
<?php
$uploadOk = 1;
```

```

if(isset($_FILES['fileToUpload'])){
    $target_file = "uploads/".basename($_FILES['fileToUpload']['name']);
    $file_name = $_FILES['fileToUpload']['name'];
    $file_size = $_FILES['fileToUpload']['size'];
    $file_tmp = $_FILES['fileToUpload']['tmp_name'];
    $file_type = $_FILES['fileToUpload']['type'];
    $parts = explode('.',$_FILES['fileToUpload']['name']);
    $file_ext=strtolower(end($parts));

    $expensions= array("jpeg","jpg","png");

    if(in_array($file_ext,$expensions) === false){
        echo "Nije dozvoljena odabrana ekstenzija dokumenta (extension not
        allowed, please choose a JPEG or PNG file.)";
        $uploadOk = 0;
    }

    if($file_size > 2097152){
        echo "Fajl mora da bude manji od 2MB";
        $uploadOk = 0;
    }

    if(file_exists($target_file)){
        echo "Fajl već postoji.";
        $uploadOk = 0;
    }

    if($uploadOk == 0){
        echo "Vaš fajl nije otpremljen";
    }
    else{
        move_uploaded_file($file_tmp,$target_file);
        echo "Success";
    }

    echo "<br>";
    echo "Izabrani fajl: ".$file_name."<br>";
    echo "Velicina fajla: ".$file_size."<br>";
    echo "Tip fajla: ".$file_type."<br>";
}
?>

```

Mora se kreirati novi direktorijum pod nazivom "uploads" u direktorijumu gde se nalazi "upload.php".  
 Dodate datoteke biće sačuvane tamo.

U primeru proverava se veličina datoteke, i da li već postoji na serveru. Ako prođe proveru, datoteka se šalje na željeno mesto na serveru.

Može iz primera da se zaključi da je za dodavanje datoteka na server neophodno korišćenje globalne promenljive `$_FILES`;

Ova promenljiva čuva sve informacije vezane za datoteka koja se dodaje na server. Ako za atribut name datoteke koja se dodaje na server kao u primeru navedemo fileToUpload, PHP će kreirati 5 promenljivih.

`$_FILES['fileToUpload']['tmp_name']` - ime dodatog fajla u privremenom direktorijumu

`$_FILES['fileToUpload']['name']` - pravo ime dodatog fajla

`$_FILES['fileToUpload']['size']` - veličina dodatog fajla

`$_FILES['fileToUpload']['type']` - tip dodatog fajla

`$_FILES['fileToUpload']['error']` - kod greške nastale pri dodavanju ove datoteke

Poslata datoteka nalazi se na privremenoj lokaciji na serveru koja se može dobiti pomoću vrednosti `$_FILES["datoteka"]["tmp_name"]`. Da bi se datoteka smestila na željeno mesto (i pod željenim imenom), potrebno je koristiti funkciju `move_uploaded_file`. Prvi argument funkcije `move_uploaded_file` je trenutna, privremena putanja do datoteke, a drugi argument je željena putanja. U primeru to je direktorijum "uploads" koji se nalazi u istom direktorijumu kao "upload.php".

## Filteri

---

Filteri se koriste za proveru ispravnosti spoljnog unosa. Podaci u neispravnom obliku mogu dovesti do pucanja veb stranice. Korišćenje filtera garantuje da će podatak biti unet u traženom obliku. Postoje 2 tipa filtera: filteri koji proveravaju ispravnost oblika i filteri koji vrše korekciju unosa.

### *Funkcija filter\_var*

PHP ima ugrađene filtere koji se ostvaruju pomoću funkcije `filter_var`. Funkcija prima 3 argumenta, prva dva argumenta su obavezna, dok je treći argument opcion.

Prvi argument je promenljiva koja se proverava. Drugi argument je naziv filtera koji se koristi. Treći argument je niz opcija koje se odnose na filter.

Rezultat funkcije je ili false ako je greska, ili filtriran podatak.

Ispravljanje stringa

U sledećem primeru koristi se funkcija `filter_var` za brisanje svih HTML etiketa iz stringa.

```
<?php
$str = "<h1>Hello world</h1>";
$str = filter_var($str,FILTER_SANITIZE_STRING);
echo $str;
?>
```

### *Proveravanje da li je unet ceo broj*

U sledećem primeru se koristi funkcija `filter_var` za proveru da li je vrednost promenljive ceo broj.

```
<?php
$int = 150;
if(!filter_var($int,FILTER_VALIDATE_INT)===false){
    echo "Jeste ceo broj". "<br>";
}
else{
    echo "Nije ceo broj". "<br>";
}
?>
```

U primeru iznad, ako se za promenljivu `$int` postavi vrednost na nula, `$int = 0` funkcija bi vratila rezultat: Nije ceo broj. Ovaj problem se rešava na sledeći način:

```
<?php
$int = 0;
if(filter_var($int,FILTER_VALIDATE_INT)===0 ||
!filter_var($int,FILTER_VALIDATE_INT)===false){
    echo "Jeste ceo broj". "<br>";
}
else{
    echo "Nije ceo broj". "<br>";
}
?>
```

### *Ispravan e-mail*

U sledećem primeru koristi se funkcija `filter_var` prvo za brisanje nedozvoljenih karaktera iz promenljive `$email`, a zatim za proveru da li je e-mail u ispravnom obliku.

```
<?php
$email = "marko.markovic@gmail.com";
$email = filter_var($email,FILTER_SANITIZE_EMAIL);

if(!filter_var($email,FILTER_VALIDATE_EMAIL)===false){
    echo "$email je ispravna e-mail adresa". "<br>";
}
else{
```

```
    echo "$email nije ispravna e-mail adresa"."<br>";  
}  
?>
```

### *Ispravan URL*

U sledećem primeru koristi se funkcija `filter_var` prvo za brisanje svih nedozvoljenih karaktera iz promenljive `$url`, a zatim za proveru da li je url u ispravnom obliku.

```
<?php  
$url = "http://itobuka.matf.bg.ac.rs/";  
$url = filter_var($url,FILTER_SANITIZE_EMAIL);  
  
if(!filter_var($url,FILTER_VALIDATE_URL)===false){  
    echo "$url je ispravan URL";  
}  
else{  
    echo "$url nije ispravan URL";  
}  
?>
```

### *Proveravanje da li je broj u određenom opsegu*

U sledećem primeru koristi se funkcija `filter_var` za proveru da li je u pitanju ceo broj i da li je izmedju 1 i 200.

```
<?php  
$num =122;  
$min = 1;  
$max = 200;  
  
if(!filter_var($num,FILTER_VALIDATE_INT,array("options"=>  
array("min_range"=>$min,"max_range"=>$max)))===false){  
    echo "Promenljiva je u traženom opsegu"."<br>";  
}  
else{  
    echo "Promenljiva nije u traženom opsegu"."<br>";  
}  
?>
```

### *Izbacivanje određenih karaktera*

U sledećem primeru koristi se funkcija `filter_var` za brisanje svih HTML etiketa iz stringa, kao i sve karaktere čiji je ASCII kod veći od 127.

```
<?php
$str = "<h1>Hello World!</h1>";
$newstr = filter_var($str, FILTER_SANITIZE_STRING, FILTER_FLAG_STRIP_HIGH);
echo $newstr."<br>";
?>
```

Videti više na <https://www.php.net/manual/en/filter.examples.php>

## Slanje mejla

---

Može biti potrebno slanje mejla kao direktan odgovor na neku akciju korisnika (npr. Logovanje na naš sajt) ili događaj koji se ponavlja u određeno vreme (npr. Newsletter).

```
<?php
$to = 'ninaradojicic@gmail.com';
$subject = 'test';
$body = '<h1>IT OBUKA</h1>';
mail($to, $subject, $body);
?>
```

Za slanje mejla pomoću php mail funkcije potrebno da je bude podešen SMTP za slanje mejla. SMTP (engl. *Simple Mail Transfer Protocol*) predstavlja osnovni protokol sloja aplikacija za elektronsku poštu, koji koristi uslugu pouzdanog transfera podataka protokola TCP. Kao i većina drugih protokola aplikacijskog sloja ima klijentsku i serversku stranu (koja se izvršava na serveru za elektronsku poštu onoga koji šalje i druga strana koja se izvršava na serveru). Pored toga SMTP host i SMTP port treba da se navedu u php.ini fajlu (lokacija se može odrediti među informacijama dobijenim pozivom:

```
<?php

phpinfo();

?>
```

U našem slučaju je to na lokaciji: C:\wamp64\bin\php\php7.3.1

Kada se SMTP server podesi, mejlovi se mogu uspešno slati pomoću php skripti.

## PHPMailer

Biblioteka koja je čest izbor i alternative klasičnoj funkciji mail().

U php.ini fajlu u liniji phpopenssl (skloniti ; sa pocetka ovog reda)

*Napomena:* Na gmail nalogu sa koga želite da pošaljete mail mora da se podesi: Allow less secure apps

## Servis Formspree

<https://formspree.io/> omogućava jednostavno dodavanje formi na stranicu sajta i obezbeđuje da svi mejlovi stignu do nas – bez PHP, JavaScript-a ili logovanja.

```
<html>
<head>
  <style>
    body {
      font-size: 24px;
    }

    .row-input {
      width: 35%;
      padding-top: 20px;
      line-height: 2em;
    }

    .row-input input,
    textarea {
      float: right;
      width: 50%;
      font-size: 24px;
    }
  </style>
</head>

<body>
  <form method="POST" action="http://formspree.io/VASAEMAILADRESA">
    <h2>Send me a message</h2>
    <div class="row-input">
      <label>Email<input type="email" name="email" placeholder="Your
email">
    </label>
    </div>
    <div class="row-input">
      <label>Your message
      <textarea name="message" placeholder="Your
message"></textarea>
    </label>
```

```
</div>
<div class="row-input">
  <label>Extra
    <input type="text" name="extra" />
  </label>
</div>
<button type="submit">Send</button>
</form>
</body>
</html>
```