

# OIPA - Osnove izrade PHP aplikacija

Marijan Šufraj

## 1 Poziv na predavanje

PHP je skriptni jezik koji se u zajednici pojavio još davne 1995-te godine. Od tada je poprilično uznapredovao do trenutne verzije, 7.3. PHP je prvo bila skraćenica za *Personal Home Page* dok danas predstavlja rekurzivnu definiciju *PHP - Hypertext Processing*. Iako ima svojih negativnih strana poput nekonzistentnih definicija funkcija i neočekivanih pretvorbi tipova podataka, to ga ne sprječava da bude jedan od najkorištenijih jezika za izradu web-aplikacija.

Jedan izvor <sup>1</sup> navodi kako otprilike 83.1% aplikacija za koje se zna u kojem su jeziku pisane, koristi upravo PHP. U tu grupu spadaju i velike web-aplikacije poput Facebooka, Wikipedije i Etsyja. Primjer veće web-aplikacije pisane u PHP-u s našeg područja je Njuškalo. Kroz vrijeme je PHP dobio podršku za objektno orijentirano programiranje, memorijski je model poboljšan i pružene su razne optimizacije kako bi izvršavanje programa bilo što brže. Zajednica je također pridonijela popularnosti PHP-a pisanjem nativnih biblioteka u C-u i korisničkih biblioteka u samom PHP-u.

### 1.1 Plan vještine

Fokus će biti na PHP 7.0+ te će stvari koje nisu podržane u starijim verzijama ili se drugačije ponašaju u odnosu na starije verzije, biti posebno naglašene. Poseban će fokus biti na sigurnosti web-aplikacija i organizaciji koda. Kako ćemo se upoznavati s određenim konceptima, tako će biti i objašnjene ranjivosti vezane uz te koncepte. Također, kako će biti potrebno, upoznat ćemo se i s potrebnim HTML tagovima iako tehnologije kao HTML/CSS/JavaScript nisu primarni fokus ove vještine. Naravno, svako naprednije korištenje navedenih tehnologija bit će posebno nagrađeno.

PHP obrađuje zahtjeve u kontekstu web-poslužitelja te ćemo se stoga prvo upoznati sa životnim ciklusom jednog takvog zahtjeva gdje ćemo dobiti malo širu sliku o radu web-poslužitelja. Upoznat ćemo se s web-poslužiteljima poput Apache Http Server, Nginx i PHP-FPM.

Kako bi si olakšali praćenje predavanja kao i sam rad na vještini, upoznat ćemo se sa sustavima za verzioniranje koda i sustavima za virtualizaciju. Kao sustav za verzioniranje bit će korišten Git kako bi lakše pratili promjene u kodu i eksperimentirali. Sustav za virtualizaciju će biti Docker i omogućit će nam bolje fokusiranje na vještinu. Pomoću sustava za virtualizaciju svi ćemo imati isto okruženje za rad što će smanjiti broj *“It Works on My Machine”* situacija <sup>2</sup>.

PHP interpreter pisan je u programskom jeziku C, stoga ćemo se prvo upoznati s memorijskim modelom PHP-a kako bi praćenje kasnijih tema bilo jednostavnije. To uključuje upoznavanje sa strukturom podataka *zval*, automatsko upravljanje memorijom i automatske pretvorbe tipova podataka.

Nakon toga ćemo se pozabaviti samom sintaksom jezika i osnovnim gradivnim elementima poput

---

<sup>1</sup><https://w3techs.com/technologies/details/pl-php/all/all>

<sup>2</sup>Vidi odjeljak 1.3

varijable, tipovi podataka i kontrole toka kao i važnijim PHP konfiguracijskim direktivama.

Nitko od nas nije savršen pa su alati za traženje pogrešaka i više nego korisni. Stoga ćemo se upoznati s alatom Xdebug koji pomaže u otkrivanju pogrešaka. Pritom ćemo se upoznati i s instalacijom nativnih PHP biblioteka. Vođeni mislju *Fool me once, shame on you; fool me twice, shame on me*, upoznat ćemo se i s automatiziranim testovima kako ne bi više puta radili isti posao. Prvo će to biti jednostavne vlastite implementacije, dok ćemo se kasnije upoznati s naprednim sustavima poput PHPUnita i phpSpeca.

Budući da se PHP koristi preko protokola HTTP, koji ne čuva stanje, upoznat ćemo se i sa sjednicama u PHP-u koje služe upravo čuvanju stanja između uzastopnih zahtjeva.

Pozabavit ćemo se i slanjem obrazaca i datoteka na poslužitelj što će obuhvatiti i komunikaciju s datotečnim sustavom.

PHP je interpreterski jezik koji je posebno dobar u radu s nizovima znakova. Stoga ćemo se malo detaljnije upoznati s obradom nizova kao i regularnim izrazima u PHP-u.

Potom uranjamo u objektno orijentirani svijet PHP-a. Upoznat ćemo se s razredima, objektima, nasljeđivanjem i ostalim relevantnim pojmovima u kontekstu PHP-a. Naglasak neće biti toliko na semantici tih pojmova koliko na samim mogućnostima PHP-a, tj. što možemo implementirati u PHP-u, a što ne.

Kako većina web-aplikacija treba neki sustav za pohranu podataka, upoznat ćemo se i sa sustavima za upravljanje bazama podataka. Sustav koji će se koristiti na vještini je MySQL, a s njim ćemo se povezati pomoću biblioteke PDO. Same SQL naredbe neće biti detaljno obrađene, već će biti pokriveno samo osnovne funkcionalnosti.

PHP nije samo programski jezik za pisanje web-aplikacija već se može pokretati i iz konzole. Pozabavit ćemo se i s tom stranom PHP-a koja uključuje stvari poput obrade argumenata skripte i obradu signala.

Pred kraj ćemo se semestra više fokusirati na organizaciju koda korištenjem pripadnih oblikovnih obrazaca. Vidjet ćemo koje su prednosti takvog pristupa i načine za korištenje drugih biblioteka kako ne bi rješavali problem kojeg je već netko riješio. Tu će nam pomoći alat Composer s kojim ćemo i zaključiti vještinu.

## 1.2 Organizacija

Vještina će se održavati kroz ljetni semestar koji traje trinaest tjedana. Svaki će tjedan biti dva bloka po dva sata predavanja. Potrebno je biti na svim predavanjima i dopuštena su najviše dva izostanka. Neprisutnost na potrebnom broju predavanja povlači pad vještine.

Svaki se ponedjeljak zadaje zadaća koju studenti moraju predati unutar tjedan dana. Studentima je dopušteno na maksimalno dvije zadaće imati nula bodova. Zadaća koja nije predana također nosi nula bodova. Kako je poseban naglasak stavljen na sigurnost, svaki je sigurnosni propust dopušteno imati samo jednom. Tako da, ako u jednoj zadaći imate ranjivost na napad XSS, svaka sljedeća zadaća na kojoj ćete imati takvu ranjivost odmah dobiva nula bodova.

Na samom kraju studenti rade svoj samostalni studentski projekt (SSP) kako bi demonstrirali stečena znanja na vještini. Dopušteno je korištenje svega obrađenog na vještini kao i naprednih stvari koje studenti sami odluče koristiti. Prilikom ocjenjivanja SSP-a studenti će trebati objasniti određene dijelove koda i pokazati razumijevanje naučenog. Nema ništa gore od *kopiraj/zalijepi* sa StackOverflowa bez razumijevanja što određeni komad koda radi.

Ako imate nejasnoća, u bilo kojem trenutku možete pitati nekog od članova osoblja vještine za pomoć

i to se toplo preporučuje.

### 1.3 Dodatne napomene

Predavanja se izvode na Linux operacijskom sustavu unutar Docker okruženja. Kako si Windows operacijski sustav i Docker ne idu kao ruka pod ruku, preporuka je instalirati Linux distribuciju po volji unutar Virtualbox okruženja i raditi unutar tog virtualnog stroja.

Na predavanjima se obradi više nego što to piše u prezentacijama (nitko ne želi prezentacije od 500 slajdova). Na žalost, neke literature poput knjige nema, već su dane poveznice u predavanjima na određene resurse na internetu. Iz istog je razloga preporuka voditi bilješke tijekom predavanja, kako bi se postigao što bolji rezultat.

### 1.4 Preuvjeti

Vještina je namijenjena svim studentima koji žele naučiti PHP i svi se mogu prijaviti. Prijave nisu ograničene samo na studente FER-a. Za sve će prijavljene studente biti održan kvalifikacijski ispit koji ima dva dijela. Prvi je dio poznavanje jezika C a drugi je dio poznavanje HTML-a i sam će termin tog ispita biti objavljen naknadno.

Kako će se raditi razni koncepti koji se uče tek na kasnijim godinama, preporuka je da vještinu upisuju studenti treće i viših godina, no to je samo preporuka.

Dodite i upoznajte se s PHP-om na vještini *Osnove izrade PHP aplikacija*.