

STATIČKO I DINAMIČKO UPRAVLJANJE WEB SADRŽAJEM

STATIC AND DYNAMIC WEB CONTENT MANAGEMENT

Barbara Kořak, Mario Tomiřa, Marko Čačić

Prethodno priopćenje

Sažetak: Upravljanje web sadržajem je kompleksan proces. Kako bi se pojedini model upravljanja što jasnije definirao potrebno je kritički sagledati tehnologije i alate za izradu web stranica. Za definiranje statičkog modela upravljanja web sadržajem koristit će se osnovne tehnologije za izradu web stranica - HTML i CSS, dok će se dinamički model definirati na primjeru CMS sustava. Upravljanje web sadržajem odnosi se na upisivanje, uređivanje i brisanje sadržaja tehnički završenog web projekta, no kako bi se sam proces mogao detaljno opisati i analizirati potrebno ga je sagledati od faze planiranja. Obzirom da u procesu izrade web stranice sudjeluje više osoba tj. stručna osoba i klijent, uz analizu i definiciju potrebno je razmotriti i odnos dizajnera/developera te klijenta/korisnika.

Ključne riječi: CMS, CSS, HTML, model upravljanja, dinamičko upravljanje, statičko upravljanje

Preliminary notes

Abstract: Web content management is a complex process. In order to define each model in detail, it is necessary to review various web design technologies and tools. Basic web design technologies- HTML and CSS - will be used to define static web content management model, while dynamic model will be defined on the example of CMS system. Web content management refers to the writing, editing and deleting of a finished web project's content. However, in order to describe and analyze it in details, it is necessary to review it from the planning phase. Given that in the process of web design more than one person is involved, i.e. an expert and a client, aside from analysis and definition it is necessary to consider the relationship between the designer/developer and the client/user as well.

Keywords: CMS, CSS, HTML, management model, dynamic management, static management

1. UVOD

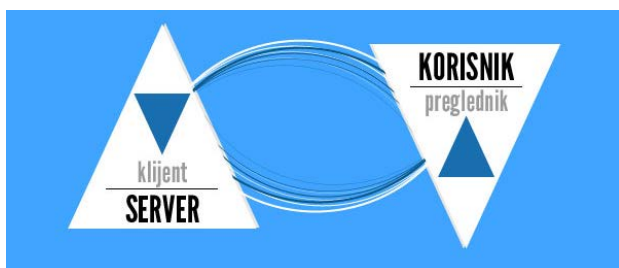
Prema osnovnoj podjeli web stranice se dijele na statičke i dinamičke pa stoga postoji i podjela prema modelu upravljanja. Model upravljanja može se definirati kao skup metoda, alata i pristupa koji tvore okolinu za upravljanje sadržajem web mjesta. U svrhu definiranja statičkog modela upravljanja web sadržajem, u ovom radu će se koristiti osnovne web tehnologije, tj. HTML i CSS, dok će se dinamički model definirati upotrebom CMS sustava.

Za analizu pojedinog modela potrebno je istaknuti i njihove prednosti te nedostatke. Problematika oko odabira odgovarajućeg modela bazira se na ukupnoj korisnosti i efikasnosti web mjesta. Točnije, unatoč tome što se u načelu upravljanje web sadržajem odnosi na ono krajnje upravljanje, odnosno upravljanje sadržajem završenog projekta, odabir i funkcionalnost odgovarajućeg modela ima utjecaj na cjelokupan proces razvoja projekta.

U radu će se najprije iznijeti definicije pojedinog modela upravljanja. Zatim će se analizirati korisnosti i usporediti funkcionalnosti, kako s tehničkog tako i s vizualnog aspekta, a zatim iznijeti zaključak temeljen na istraživanju i analizi modela upravljanja web sadržajem.

2. STATIČKO UPRAVLJANJE WEB SADRŽAJEM

Do općeg prihvaćanja web 2.0 modela te pripadajućih tehnologija, odnosno dinamičkog modela upravljanja, većina web stranica bila je statička. U svojim počecima, web stranice su činile samo HTML datoteke, koje su isključivo HTML kodom opisivale web pregledniku organizaciju sadržaja koji se prikazuje. Zahtjevom dizajnera da se HTML jezik ne koristi i za vizualno oblikovanje, razvio se CSS standard, što je omogućilo odvajanje informacije o strukturi web stranice od informacija o njenom izgledu. [1] Mogućnosti vizualnog oblikovanja web stranica pomoću modernog CSS3 jezika, u odnosu na prve verzije značajno su napredovale, no u osnovi koncepcija upravljanja ostala je ista. Postoje određene razlike u procesu izrade, no konceptualno proces je identičan onome iz ranih dana weba. Kodovi se grupiraju u datoteke i postavljaju na web poslužitelj, koji osluškuje vanjske upite i dostavlja zatražene dokumente HTTP protokolom. Za svaku naknadnu promjenu, potrebno je modificirati željene kodove te izmijenjene datoteke ponovno postaviti na web poslužitelj kako bi učinjene promjene postale vidljive vanjskim korisnicima.



Slika 1. Komunikacija u okruženju statičkog web mjesta

S tehničkog aspekta, statičko upravljanje web sadržajem odnosi se na ukupan proces izrade i ažuriranja web stranice prema načelima statičkog okruženja, pri čemu se koriste samo osnovne web tehnologije, HTML i CSS. Statičko okruženje karakterizira nedostatak praktičnog modela kontrole sadržaja, odnosno ne postoji nikakav oblik korisničkog sučelja za upravljanje web sadržajem, već se upravlja isključivo verzioniranjem datoteka. Statičko web mjesto može se još opisati i kao skup povezanih HTML datoteka s predodređenim, fiksno definiranim sadržajem koji ostaje nepromijenjen neovisno o korisnikovim zahtjevima prilikom posjete. Za ažuriranje i uređivanje statičkog web mjesta potrebne su stručne vještine, a samim time što su sadržaj i kod isprepleteni u datotekama, klijent bez stručnog znanja nema mogućnost samostalnog mijenjanja sadržaja. [1, 2] S vizualnog aspekta, statičko web mjesto ne nameće gotovo nikakva ograničenja u kontekstu vizualnog oblikovanja.

2.1. Povijesni pregled

Povijesni razvoj HTML standarda započinje 1989. godine kada Berners-Lee po prvi puta spominje pojam HTML. Šest godina nakon toga, verzija HTML 2.0 postaje prvi službeni standard. Sljedećih godina dolazi do manjih preinaka standarda kao i do standardiziranja posebnog stilskog jezika, CSS-a. Godine 1997. prihvaća se HTML 4.0, a 1998. CSS 2 standard, koji su zatim u upotrebi sljedećih desetak godina. Novi HTML 5.0 standard objavljen je 2008. kao javni dokument skice, no u širokoj je upotrebi zajedno s CSS 3 standardom već od 2012. godine. [12]

Razvoj HTML i CSS standarda imao je izravan utjecaj i na okolinu statičkog upravljanja web sadržajem, odnosno na strukturu datoteka kojima se upravlja. Analizirajući povijesni razvoj web tehnologija, može se zaključiti kako je u vrijeme stagnacije razvoja opisnih jezika, HTML-a i CSS-a, istovremeno započeo intenzivan razvoj CMS sustava, odnosno dinamičkih principa upravljanja web okolinom.

2.2. Vrste i funkcionalnosti statičko upravljanih web mjesta

Prema McLuhanu, stari medij postaje sadržaj novog. Ta tvrdnja može se primijeniti i u slučaju statičkog web okruženja, a razlog tome jest što se u kontekstu statičkih stranica prvenstveno govori o usporedbi s brošurama (eng. brochureware). [1] Termin web brošure polazi od prvobitne primjene weba u svrhu digitalne reinterpretacije informacija objavljenih u tiskanim

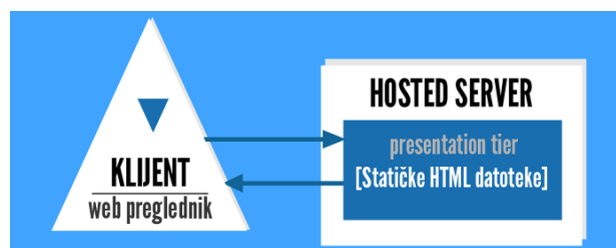
brošurama, što su činile mnoge organizacije i tvrtke prisutne na webu u to vrijeme. Termin se zadržao i kasnije pa se stoga danas koristi kao sinonim za statičku stranicu. Možemo zaključiti da su web brošure vrsta web stranica kojima je cilj pružiti informacije na gotovo identičan način kao tiskana brošura. Osim web brošura, neke od najčešće upotrijebljenih vrsta statičkih web stranica su web posjetnice i portfolio stranice.

S aspekta funkcionalnosti, prednost statičkih web stranica i pripadajućeg modela upravljanja ogleđa se u jednostavnosti upravljačke okoline te u mogućnostima vizualnog oblikovanja. Točnije, obzirom da je osnovna karakteristika statičkih stranica jednosmjerna komunikacija prema korisniku, fokus je na samoj prezentaciji informacija.

2.3. Hosting statičkog web mjesta

Nakon finaliziranja projekta statičkog web mjesta, potrebno ga je postaviti na web poslužitelj kako bi postalo dostupno krajnjim korisnicima.

U okviru statičkog modela upravljanja, pojam hostinga svodi se na centralizirano posluživanje statičkih HTML dokumenata, na zahtjev korisnika. Proces započinje postavljanjem datoteka na poslužitelj, najčešće pomoću FTP klijenta. Slijedi konfiguriranje poslužitelja, odnosno povezivanje s registriranom domenom kroz DNS sustav, nakon čega je korisniku omogućen pristup. Korisnik kroz web preglednik zahtjeva prikaz web mjesta pretragom njegove simboličke domene, a server zatim izvršava funkciju isporuke traženih datoteka. Princip rada hostinga vidljiv je na slici 2.



Slika 2. Shema komunikacije kod hostinga statičkog web mjesta

2.4. Prednosti

Osnovna prednost statičkog upravljanja jest razina kontrole, što se prvenstveno odnosi na potpunu kontrolu datoteka i sadržaja uključenih u web mjesto te kontrolu potrošnje resursa web poslužitelja jer statičke stranice nije potrebno programski obrađivati na strani poslužitelja.

Ako se uzme u obzir teorija da korisnik odustaje od posjeta stranici ukoliko njeno učitavanje traje duže od samo tri sekunde, vrijeme je od iznimne važnosti. [4] Statičke stranice nemaju bazu podataka iz koje se učitava pripadajući sadržaj pa je stoga brzina njihova učitavanja zamjetno veća. Obzirom na značaj navedene prednosti, preporučljivo je shvatiti ju kao primarnu vodilju pri projektiranju statičkog web mjesta, kako bi vrijeme učitavanja bilo što kraće, a samim time i korisničko iskustvo bolje.

Statičke stranice također nije potrebno niti ažurirati sigurnosnim zakrpama, što je posebno važno u okviru dinamičkog web okruženja kako bi se osigurala visoka razina sigurnosti i raspoloživosti. Nadalje, fleksibilnost izrade statičkih stranica je neograničena, što ujedno znači i potpunu kontrolu nad obje strane, one administrativne i one prezentacijske. [5]

Za izradu statičkog web mjesta potrebne su vještine kodiranja pomoću HTML i CSS opisnih jezika, a po izboru i nekih od programskih, primjerice JavaScript. Razvojem vještina kodiranja raste i mogućnost izrade kompleksnijih stranica, posebice ako su u pitanju HTML5 i CSS3 standardi koji svojim svojstvima pružaju mogućnost izrade izrazito kompleksnih stranica bez dodatnog skriptiranja. [12]

Hosting je jednostavniji i brži obzirom da je za uspostavu web mjesta potrebno samo registriranje i konfiguracija simboličke domene te postavljanje datoteka na odabrani web poslužitelj. Također, migracija web mjesta između poslužitelja je jednostavnija jer nema baze podataka, administracijskih sustava niti ostalih dodataka koje je potrebno rekonfigurirati.

Statički model upravljanja web sadržajem ima mnogo prednosti nad dinamičkim, no te prednosti najbolje se daju iskoristiti u specifičnim situacijama. Točnije, prilikom faze planiranja potrebno je uzeti u obzir sve ključne parametre i prema tome utvrditi odgovara li princip statičkog modela upravljanja ciljanom slučaju. Neki od ključnih parametara analize određuju se prema opsegu sadržaja, uvjetu interakcije, namjeni i cilju te financijskim i vremenskim resursima. Statički model upravljanja prikladan je izbor u slučaju kada je opseg projekta otprilike do deset stranica, ako će sadržaj biti pouzdan izvor informacija duži vremenski period te ako se ne zahtijevaju oblici interakcije kao što su komentiranje, registriranje korisnika ili slično. U slučaju da se projektu ipak želi nadodati barem neki oblik dinamičnosti, odnosno interakcije s korisnicima, to je moguće ostvariti pomoću programskih jezika, primjerice JavaScript-a na strani klijenta ili PHP-a na strani poslužitelja.

Unatoč tome što trenutno na tržištu prevladava upotreba dinamičkih stranica, statičke stranice i statičko upravljanje web sadržajem ostat će temelj web dizajna. U krajnjem slučaju, kako bi znali upravljati u dinamičkom okruženju, najprije je potrebno usavršiti statički model upravljanja te izradu statičkih web stranica. S aspekta vizualnog oblikovanja, prednost statičkog modela jest ta da ne postoji kontrolirano okruženje unutar kojeg se upravlja oblikovanjem. Proces se svodi isključivo na povezivanje i upotrebu CSS stilskog jezika, za razliku od dinamičkog modela gdje se web dizajner mora prvo prilagoditi korisničkom sučelju, a zatim birati između nekoliko različitih načina oblikovanja web stranice.

2.5. Nedostaci

Jedan od glavnih nedostataka statičkog modela upravljanja jest njegova kompleksnost, koja raste proporcionalno s opsegom projekta, odnosno povećanjem količine datoteka uključenih u web mjesto smanjuje se efikasnost u vidu brzine učitavanja i jednostavnosti upravljanja.

Nadalje, odabir statičkog modela upravljanja za klijenta znači nemogućnost samostalne izmjene web sadržaja, odnosno ovisnost o stručnoj osobi. Nedostatak se manifestira i kod prebacivanja odgovornosti s jedne na drugu stručnu osobu s obzirom na to da je potrebno ponovno upoznavanje s kodom i određeni vremenski period da bi druga osoba posve savladala strukturu datoteka i koda. [6]

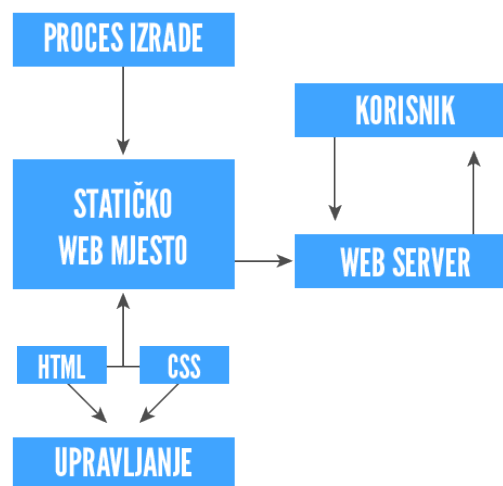
Kod ažuriranja nakon finalizacije projekta, najveći se nedostatak može opisati kroz primjer promjene telefonskog broja. Na primjer, web mjesto se sastoji od sveukupno osam stranica od kojih svaka sadržava podnožje unutar kojeg se nalazi telefonski broj. Kad klijent zadaje zahtjev za promjenom pa makar i samo jedne brojke, dizajner mora u svakoj zasebnoj HTML datoteci, u ovom slučaju njih osam, pronaći kod za podnožje i u svakoj od njih promijeniti telefonski broj. Ista situacija se događa i s ostatkom sadržaja, obzirom da ne postoji nikakav oblik korisničkog sučelja kroz koji je moguće promijeniti samo sadržaj, bez intervencije u sam kod. [7]

Proces transformacije statičkog web mjesta u dinamičko je načelno logičan slijed, posebice ako se radi o slučaju gdje određena organizacija želi proširiti svoje poslovanje na webu te poboljšati komunikaciju s korisnicima. Unatoč tome, ova opcija razmatra se tek nakon što su s klijentom utvrđeni svi parametri kako ne bi došlo do nepotrebnog troška.

Kao što je definirano ranije, statička web mjesta u pravilu služe za jednosmjernu komunikaciju s korisnikom, odnosno funkcioniraju po principu brošure. Nadalje, obzirom na potrebe modernih korisnika u smislu interakcije, mnogi vlasnici statičkih web stranica se odlučuju na komunikaciju putem društvenih mreža, što u krajnjem slučaju znači da vlasnik mora balansirati svoju prezentaciju na dva različita servisa.

Svi nedostaci statičkog modela upravljanja ponajviše se manifestiraju ukoliko su zanemareni savjeti i ključni parametri prilikom faze planiranja. Izrada statičkog web mjesta u načelu je jeftinija u odnosu na dinamičko, no neopreznim planiranjem vrlo brzo i jednostavno može doći do neočekivanih troškova.

2.6. Statički model: Vizualni prikaz

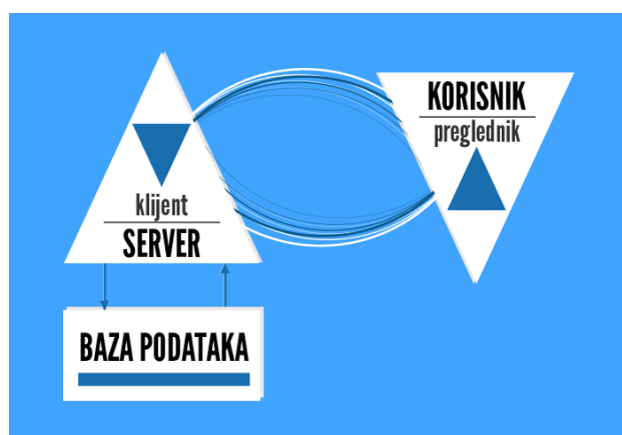


Slika 3. Shematski prikaz razvoja i upravljanja statičkog web mjesta

Statičko web mjesto primarno se sastoji od HTML i CSS datoteka, pomoću kojih se vrši statičko upravljanje web sadržajem. Nakon završetka izrade statičkog web mjesta, glavni direktorij se postavlja na web poslužitelj (eng. server). Web poslužitelj i korisnik komuniciraju putem web preglednika s korisnikove strane i domene koja ostvaruje poveznicu s web mjestom.

3. DINAMIČKO UPRAVLJANJE WEB SADRŽAJEM

Dva su osnovna pojma koja opisuju dinamičko upravljanje web sadržajem a to su: korisničko sučelje i baza podataka. Komunikacija s korisnikom odvija se jednako kao i kod statičkih web mjesta, a shema je prikazana na Slici 4. [1]



Slika 4. Komunikacija kod dinamičkog web mjesta

Glavna karakteristika po kojoj se dinamički model upravljanja prepoznaje jest to što se pokreće pomoću specijaliziranog sustava za upravljanje, odnosno CMS sustava (eng. Content Management System). CMS je sustav kojemu je primarna svrha dinamičko ažuriranje sadržaja na webu, a sastoji se od dva dijela: CMA i CDA. CMA, odnosno aplikacija za upravljanje sadržajem (eng. Content Management Application) označava korisničko sučelje putem kojeg se izvodi sveukupno upravljanje, dok CDA označava aplikaciju za dostavljanje sadržaja (eng. Content Delivery Application) kojoj je svrha kompiliranje svih informacija potrebnih za pokretanje i izvođenje operacija u sklopu procesa upravljanja sadržajem. Ove dvije aplikacije su još jedna karakteristika dinamičkog upravljanja s obzirom na to da one zapravo omogućuju odvajanje sadržaja od izvornog koda, za razliku od statičkog modela gdje su i sadržaj i kod isprepleteni unutar jednog dokumenta. [8]

Dinamički model upravljanja web sadržajem u svojem najjednostavnijem obliku može jednostavno potpuno imitirati statičku stranicu s dodanom prednošću korištenja korisničkog sučelja i jednostavnog ažuriranja. [1] Ipak, s obzirom na to da su CMS sustavi koji se najčešće koriste izrazito kompleksni i sadržavaju mnogobrojne funkcionalnosti, dinamički model upravljanja najbolje funkcionira kod većih projekata. Točnije, kod projekata koji zahtijevaju često ažuriranje,

arhiviranje podataka, baze korisnika, a samim time i sudjelovanje više osoba na razvoju projekta. [1]

CMS sustav je specijalizirani softver ili aplikacija koja omogućuje izradu i upravljanje web mjestom, a neke od osnovnih karakteristika su: unos i objava sadržaja putem WYSIWYG uređivača, upravljanje multimedijalnim sadržajem, arhiviranje podataka, upravljanje predlošcima, autorizirani pristup, itd. [9]

3.1. Povijesni razvoj CMS sustava

Razvoj CMS sustava može se podijeliti u tri dijela: upravljanje – M (eng. Management), sustavi – S (eng. System) i sadržaj – C (eng. Content). Upravo tim redoslijedom kretao se i proces razvoja, od jednostavnih sustava za upravljanje pa sve do kompleksnih CMS sustava s mnogobrojnim svojstvima, funkcionalnostima i modulima. [10]

U početku razvoja CMS sustava, fokus je bio na ostvarenju jednostavnijeg upravljanja, odnosno ostvarenju mogućnosti dinamičkog upravljanja web sadržajem u tada još uvijek statičkoj razvojnoj okolini. Nakon što je usavršen koncept upravljanja, cilj je bio tehnički stvoriti što efikasniji informacijsko-komunikacijski sustav. Tek nakon što je razvoj ta dva dijela dotaknuo određenu granicu u vidu razvoja sveukupne web tehnologije, fokus se usmjerio prema sadržaju. [1] Nakon određenog vremenskog perioda i napretka u razvoju sustava, industrija se podijelila na EDMS (eng. Enterprise Document Management System) i WCMS (eng. Web Content Management System) odnosno na sustave za upravljanje dokumentacijom i sustave za upravljanje web sadržajem. Daljnja podjela je na sustave otvorenog ili zatvorenog koda, komercijalne, enterprise, itd. Broj CMS sustava neprestano raste, a samim time rastu i kategorije po kojima se dijele. [10]

Problematika oko pitanja jesu li sustavi otvorenog koda prikladni za komercijalnu upotrebu riješila se stvaranjem CMS alata treće generacije sa svojstvom priklanjanja dizajnu, a ne kodiranju, odnosno tehnici. Neka od ključnih svojstava takvih sustava su: platforme s uključenim hostingom, preprodavanje isključivo od strane dizajn agencija, integracija s CRM-om, bazom podataka, web trgovinom, elektroničkom poštom, modularni razvoj koji minimalizira potrebno kodiranje za integraciju, SaaS (eng. Software as a Service) model cijena, itd. Usmjeravanje razvoja na dizajn dovelo je i do podjele CMS industrije na razvoj sustava s fokusom na profesionalni web dizajn te one otvorenog koda, namijenjene programerima i tehnološki iskusnijim blogerima. Ključan korak u razvoju CMS sustava je implementacija administracijskog sučelja u proces upravljanja sadržajem, što je omogućilo brzo i jednostavno uređivanje željenih web stranica bez izravne intervencije u njihov izvorni kod. [10]

3.2. Vrste i funkcionalnosti CMS sustava

CMS sustavi se ugrubo dijele na one zatvorenog i one otvorenog koda, a zatim i na komercijalne te besplatne. Karakteristika sustava zatvorenog koda jest ta da su takvi

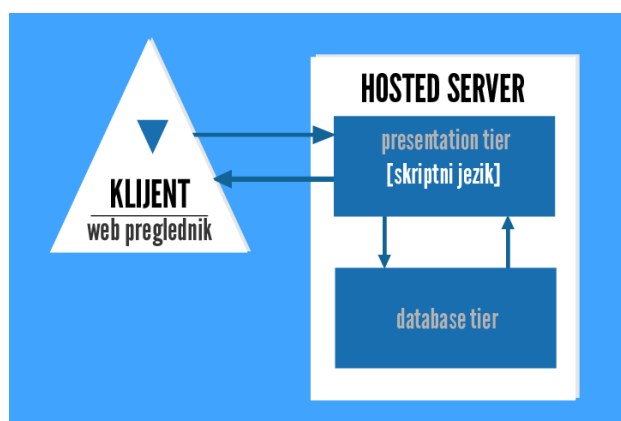
sustavi u vlasništvu neke organizacije, a korisnici prilikom preuzimanja tj. kupnje takvih sustava nemaju pristup izvornim datotekama. S druge strane postoje sustavi otvorenog koda, što znači da po preuzimanju korisnik ima pristup izvornim datotekama i kodovima koje može mijenjati i personalizirati. Nadalje, postoje još i tzv. prilagođeni sustavi, odnosno CMS sustavi koji se programiraju prema zahtjevima klijenta i u načelu ne postoji komercijalna prodaja. Također postoje i enterprize sustavi, odnosno sustavi namijenjeni organizacijama koje se bave velikim količinama informacija, a kojima će se upravljati unutar CMS sustava.

Inovacija na CMS tržištu su sustavi bazirani na datotekama (eng. File-based CMS). Glavna karakteristika ovakvih sustava jest ta da nema baze podataka, što se koristi i kao drugi naziv (eng. Non-Database CMS). [11]

3.3. Hosting dinamičkog web mjesta

Karakteristika dinamičkih web mjesta jest da korisnik prilikom učitavanja ne prima izravno zatražene HTML dokumente, već se oni prvo generiraju pomoću *server-side* programskih jezika, iz pripadajuće baze podataka u koju su prethodno spremljeni potrebni sadržaji. [3]

Komunikacija s korisnikom odvija se između web preglednika, baze podataka i web poslužitelja, što je vidljivo na slici 5. Poslužitelj se sastoji od dva segmenta, odnosno „presentation tier-a“ i „database tier-a“. „Presentation tier“ se u ovom slučaju sastoji od *server-side* programskog jezika, što je najčešće PHP, kao kod primjerice Wordpress sustava. Skriptni jezik koji se kompilira u prezentacijskom dijelu povezan je s bazom podataka iz koje se čita potreban sadržaj. Nakon komunikacije s bazom podataka, web poslužitelj svojim skriptnim jezikom zatim isporučuje podatke web pregledniku i time omogućuje konačnu prezentaciju zatražene stranice. [3]



Slika 5. Shema komunikacije kod dinamičkog web mjesta

3.4. Prednosti

Upotreba CMS sustava donosi mnogobrojne prednosti, što se prvenstveno odnosi na mogućnost dinamičkog upravljanja web sadržajem unutar kontrolirane okoline, odnosno korisničkog sučelja.

Sučelja takvih sustava najčešće se programiraju tako da je vrijeme upoznavanja korisnika sa sučeljem minimalno. Iz navedenog proizlazi da i osobe s osnovnim znanjem rada s računalnim programima mogu unositi potrebne sadržaje i ažurirati web željene web stranice. [1]

Kod statičkih web stranica sadržaj i HTML kod isprepleteni su u jednoj datoteci, dok su kod CMS sustava odvojeni. [1] Prema primjeru Wordpress-a, datoteke koje pokreću sustav nalaze se u direktoriju na tvrdom disku poslužitelja, a sučelju sustava pristupa se putem web preglednika. Upravljanje web sadržajem, kao i vizualno oblikovanje izvodi se kroz administracijsko sučelje, prilikom čega se sadržaj sprema u bazu podataka, a vizualni predlošci se poslužuju iz glavnog direktorija CMS-a.

Kolaboracija je jedan od važnih faktora pri odabiru modela upravljanja, posebice ako se radi o kompleksnijem web mjestu. CMS sustavi svojim modulom za autorizirani pristup omogućavaju kreiranje korisničkih računa s različitim ulogama, koji omogućavaju svakom autoru zaseban profil pomoću kojeg sudjeluju u upravljanju. Nadalje, upravljanje sadržajem obogaćeno je mnogobrojnim svojstvima i opcijama, kao što su arhiviranje podataka, kategorizacija i indeksiranje, ručno podešavanje vremena objave, itd. Arhiviranje podataka može se predefinirati po datumu, mjesecu, godini itd., što je od velike važnosti za sustave sa svakodnevnim ažuriranjem i velikim količinama arhiviranih podataka. Moduli kao što su bilteni, kalendari, web trgovina, forumi, korisničke baze i slično mogu biti unaprijed integrirani u sustav ili se dodaju s nekoliko jednostavnih koraka unutar korisničkog sučelja.

S aspekta vizualnog oblikovanja, većina sustava ima omogućenu opciju kreiranja i postavljanja vlastitog predloška ili korištenje gotovih, bilo komercijalnih ili besplatnih.

Što se strukture tiče, kod Wordpress-a se predložak programira pomoću PHP jezika, a najosnovnija struktura sastoji se od zaglavlja, sadržaja i podnožja. Ovakav način strukturiranja pogoduje dinamičkom ažuriranju s obzirom na to da je primjerice promjenu naslova u zaglavlju moguće učiniti na samo jednom mjestu, a promjena se primjenjuje na sve postojeće stranice unutar web mjesta.

Sveukupan proces izrade web mjesta upotrebom dinamičkog modela upravljanja puno je kompleksniji od onog sa statičkim modelom. I dok je kod statičkog web mjesta nakon razmatranja ključnih parametara dovoljno samo utvrditi željeni dizajn, kod dinamičkog modela potrebno je odrediti i koja vrsta CMS sustava je prikladna za projekt. Za dizajnera/developera odabir dinamičkog modela upravljanja znači da nakon finalizacije projekta nije odgovoran za ažuriranje sadržaja i taj dio može prepustiti klijentu, dok za klijenta znači da je nakon upoznavanja sa sustavom sposoban za samostalni unos i ažuriranje. Time se dolazi i do prednosti dinamičkog modela u vidu dodavanja korisničkih računa s ulogama pa tako dizajner/developer može ostaviti korisnički profil administratora i nakon finalizacije, a klijentu izraditi profil s kojim je osiguran pristup samo dijelu za upravljanje sadržajem. S tehničkog aspekta, dinamičko web mjesto samo po sebi pruža mogućnosti interakcije s korisnikom, a pojačanje

dvosmjernu komunikaciju moŹe se postići dodavanjem modula i dodatnih funkcionalnosti.

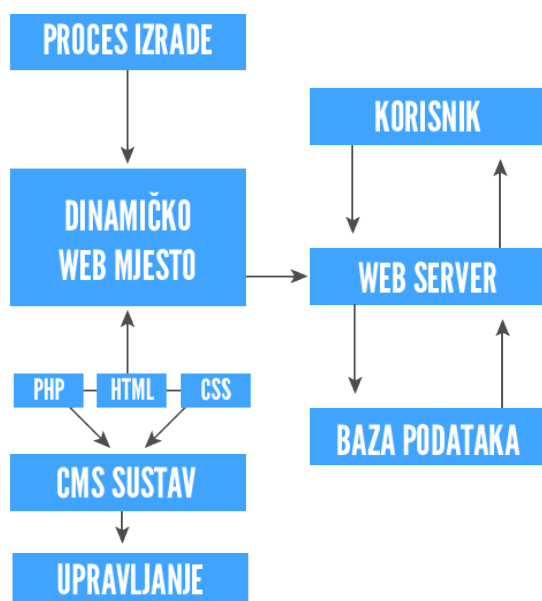
3.5. Nedostaci

Za manje projekte dinamiĉki model upravljanja uglavnom nije isplativ uzevši u obzir kompleksnost sustava. Također, iako ovisi o opsegu projekta, u naĉelu su za izradu dinamiĉkog web mjesta potrebni veći vremenski i financijski resursi, kako za vrijeme izrade, tako i za podršku nakon finaliziranja projekta. Također je u trošak potrebno uraĉunati upoznavanje klijenta sa sustavom.

Kod besplatnih sustava potrebno je samostalno osigurati hosting koji je u naĉelu financijski skuplji nego kod statiĉkog modela s obzirom na to da je potrebno više prostora na poslužitelju, povezivanje s bazom podataka, itd.

Nedostaci vezani uz dinamiĉki model upravljanja web sadržajem uglavnom se manifestiraju ili zbog nepridržavanja uputa prilikom faze planiranja ili nestručnim upravljanjem. S vizualnog aspekta, najveći nedostatak jest nepotpuna kontrola nad ukupnim vizualnim oblikovanjem, odnosno dizajnom web mjesta u odnosu na statiĉki model. Dok kod statiĉkih web mjesta dizajner ima potpunu kontrolu nad svim vizualnim elementima i datotekama kojima upravlja putem HTML i CSS jezika, kod dinamiĉkih web mjesta je to ipak drugaĉije i kompleksnije. Ukoliko dizajner Źeli vizualno oblikovati dinamiĉki upravljano web mjesto na platformi kao što je Wordpress, uz postojeće znanje HTML i CSS jezika potrebno je poznavanje i programskog jezika PHP, koji posreduje između server-side komponenti sustava.

3.6. Dinamiĉki model: Vizualni prikaz



Slika 6. Shematski prikaz razvoja i upravljanja dinamiĉkog web mjesta

Jedan od mogućih shematskih prikaza dinamiĉkog web mjesta nalazi se na Slici 6. Nakon procesa izrade nastaje dinamiĉko web mjesto, koje je izrađeno pomoću

PHP, HTML i CSS jezika, dok se upravljanje vrši putem CMS sustava, odnosno korisniĉkog suĉelja. Komunikacija s korisnikom odvija se između web preglednika, web poslužitelja i baze podataka.

4. ZAKLJUĀAK

Nakon analize korisnosti i nedostataka oba modela upravljanja sadržajem web mjesta, dolazi se do zakljuĉka kako se niti jedan od njih ne moŹe okarakterizirati kao generalno bolji ili lořiji, već na temelju ulaznih parametara i detaljnog razmatranja svakog pojedinog projekta valja odabrati prikladan model.

Na temelju istraŹivanja moŹemo zakljuĉiti kako je budućnost upravljanja web sadržajem u kombiniranju tehnologija. Već sad je primijećeno kako developeri sve više rade alate i web aplikacije koje kombiniraju prednosti pojedinog osnovnog modela upravljanja i stvaraju nove tehnologije, sustave, metode i pristupe. Daljnja istraŹivanja moguće je provesti tako da se detaljnije analiziraju i usporede novi modeli dinamiĉkih web mjesta te obrade i oni modeli koji će se tek razviti u skladu s trendovima u produkciji web mjesta i statiĉkim i dinamiĉkim upravljanjem web sadržajem.

5. LITERATURA

- [1] Lance Scaife-Elliot: „Master in Design (Research) – The transition from static to dynamic web design“, University of Technology Sydney, Sydney, 2013
- [2] NextGen Websites: The Differences between Static and Dynamic Websites, <http://www.next-generation-websites.com/the-differences-between-static-and-dynamic-websites/>, 2012, (Dostupno: 10.06.2014.)
- [3] European Commission – Digit: IS Hosting Services, Infrastructure Services Provision DIGIT Directorate, http://publications.europa.eu/tenders/our/documents/ao_10463/ao10463_annex_19_ishs_hosting_services_ec_digit_en.pdf, 2012, (Dostupno: 10.06.2014.)
- [4] instantShift; Calvin Sellers: Why Good Website Design is So Important: The Stats and Figures to Prove It [Infographic], <http://www.instantshift.com/2013/09/02/importance-of-effective-website-design-infographic/>, (Dostupno: 10.06.2014.)
- [5] Speckyboy; Ada Ivanova: Good Old Static HTML Sites Aren't Dead Yet. Should They Be?, <http://speckyboy.com/2012/02/22/good-old-static-html-sites-arent-dead-yet-should-they-be/>, 2012, (Dostupno: 10.06.2014.)
- [6] Modern Dignity: Advantages And Disadvantages Of A Static Website, <http://www.moderndignity.com/advantages-and-disadvantages-of-a-static-website/>, 2012, (Dostupno: 10.06.2014.)
- [7] Udemy; Kasia Mikoluk: Old Static Content vs New Dynamic Websites And Data,

- <https://www.udemy.com/blog/static-vs-dynamic/>, 2013, (Dostupno: 10.06.2014.)
- [8] SearchSOA; Margaret Rouse: Essential Guide: Web Content Management Systems Roundup, <http://searchsoa.techtarget.com/definition/content-management-system>, 2011, (Dostupno: 10.06.2014.)
- [9] Radarearth; Kabir Bedi: 10 Features Of A Great CMS Solution, <http://radarearth.com/content/10-features-great-cms-solution-0>, (Dostupno: 10.06.2014.)
- [10] Contegro: A (brief) history of CMS development, http://www.contegro.com/info-center/designers-blog/blog-article/_thread_/a-brief-history-of-cms-development, 2012, (Dostupno: 10.06.2014.)
- [11] GCO Studio; Jose Guerra: No Database CMS or blog, <http://gcostudios.com/2013/no-database-cms-or-blog/>, 2013, (Dostupno: 10.06.2014.)
- [12] Vinko Štajdohar, S.; Tomiša, M.; Valdec, D.: Oblikovanje web stranica primjenom CSS-a 3, Tehnički glasnik, 7 (2013), 4; 399-404, 2013

Kontakt autora:

Barbara Košak, bacc. ing. techn. graph.

e-mail: bakosak@unin.hr

doc.dr.sc. Mario Tomiša

Sveučilište Sjever

Trg. dr. Žarka Dolinara 1, 48000 Koprivnica

e-mail: mario.tomisa@unin.hr

Marko Čačić, mag. ing. techn. graph.

Sveučilište Sjever

Trg. dr. Žarka Dolinara 1, 48000 Koprivnica

e-mail: marko.cacic@unin.hr