

UPOTREBLJIVOST PROGRESIVNE VEB APLIKACIJE**THE USABILITY OF THE PROGRESSIVE WEB APPLICATIONS**Dejan Predojević, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U radu su predstavljene progresivne web aplikacije kao i njihova istorija razvoja od prvog pominjanja ovog koncepta. Date su osnovne karakteristike, prednosti i mane progresivnih web aplikacija. Takođe su detaljno opisane tehnologije i korišćeni alati potrebni za razumevanje progresivnih web aplikacija. Prikazana je sve veća potreba za mobilnim rešenjima postojećih web sajtova, kao i problem razvijanja više od jedne aplikacije zbog podrške na različitim uređajima koji uzrokuje velika finansijska i vremenska ulaganja. Predstavljena je tržišna potreba koja je dovela do razmišljanja o jedno-aplikacijskom rešenju koje progresivne web aplikacije obezbeđuju. Pored navedenog opisane su i razlike između web, nativnih i progresivnih web aplikacija čija popularnost neprestano raste. U cilju demonstracije napravljena je aplikacija namenjena za rad u biblioteci. Aplikacija je razvijena uz pomoć Angulara i Ionika, a poseban akcenat je stavljen na prikazivanje mogućnosti da se od web aplikacije napravi progresivna web aplikacija, koja će moći da se koristi i na mobilnim uređajima.

Ključne reči: PWA, mobilna, aplikacija, web

Abstract – The paper presents progressive web applications as well as their history of development since the first mention of this concept. Basic features, advantages and disadvantages of progressive web applications are given. The technologies and tools used to understand progressive web applications are also described. The growing need for mobile solutions for existing websites is shown, as well as the problem of developing more than one application due to support on different devices, which causes large financial and time investments. A market need was presented that led to the idea of a one-app solution that progressive web applications provide. Differences between web, native and progressive web applications, whose popularity is constantly growing, are also described. For the purpose of demonstration, an application intended for work in the library was created. The application was developed with the help of Angular and Ionic, and a special emphasis was placed on showing the possibility of making a web application a progressive web application, which can also be used on mobile devices.

Keywords: PWA, mobile, appwlication, web

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragan Ivetić, red.prof.

1. UVOD

U proteklih nekoliko decenija tehnološki razvoj doveo je do velikog napretka u performansama računara kao i do revolucionarnih rešenja u razvoju mobilnih uređaja. Web aplikacije današnjice napravile su veliki iskorak ka boljem korisničkom iskustvu i dovele do toga da se sve više teži ka što boljem prilagođavanju aplikacija krajnjim korisnicima. Sve navedeno podstaklo je ljude da razmišljaju o jedno-aplikacijskom rešenju koje će moći da se koristi na različitim operativnim sistemima. Jedan kod koji će moći da se instalira na više različitih uređaja, jedan dizajn prilagođen da pruža najbolje moguće korisničko iskustvo, jedan programer koji će da razvija više aplikacija kucajući samo jedan kod.

U radu su predstavljene progresivne web aplikacije, kao i implementacija aplikacije za podršku rada biblioteke koja će biti korištena kao web i mobilna aplikacija. Predstavljena je istoriska, tehnološka i tržišna potreba za upotrebom progresivnih web aplikacija. Takodje je dat opis progresivnih web aplikacija, njihove karakteristike kao i prednosti i mane ovog pristupa.

2. PROGRESIVNE WEB APLIKACIJE

Progresivne web aplikacije su aplikacije koje predstavljaju kombinaciju najboljih osobina web i nativnih aplikacija. Pišu se u web tehnologijama koristeći standardne šablone koji im to omogućavaju. Progresivne web aplikacije prave vezu između web sajtova i mobilnih aplikacija, koja do njihove pojave nije bila moguća.

2.1. Istorija progresivnih web aplikacija

Iako su se progresivne web aplikacije zvanično pojavile 2015. godine, prvi put se spominju 2007. godine od strane Apple CEO Steve Jobs-a, koji je tada predstavio revolucionarno rešenje, poznato pod nazivom Iphone. Skoro celu deceniju ova ideja je ostala zanemarena i to vreme pripada velikoj ekspanziji nativnih aplikacija koje su dominirale u mobilnom svetu. Pojavom promenljivog web dizajna, web stranice su postale mnogo fleksibilnije i njihov prikaz na mobilnim uređajima postao je drastično bolji.

Progresivne web aplikacije su čekale svoj momenat i u 2015. godini programer kompanije Chrome, Alex Russel i dizajner Frances Berriman su prvi put javno predstavili progresivnu web aplikaciju koja je omogućavala mnogo bolje iskustvo rada na uređajima a pritom se zasnivala na samo jednom baznom kodu. Predstavljeno rešenje se bazira na Apple-ovoj ideji, koja je unapređena sa novim idejama koje su pojavile do 2015. godine.

Dva velika rivala, Google i Microsoft, čija trka u svetu tehnologije traje decenijama našli su zajednički jezik i postali veliki pobornici ideje o progresivnim web aplikacijama. Prvi koji su usvojili ovaj standard u svetu web-a bili su Chrome OS, Android i Windows. Iako je Microsoft morao da se odrekne svoje univerzalne Windows platforme, kako bi napravio prostora da se posveti progresivnim web aplikacijama u kojima je video veliki potencijal [1].

2.2. Tržište

Pojava progresivnih web aplikacija imala je veliki uticaj na promenu na tržištu aplikacija i dovela do toga da svi oni koji ne razmišljaju u ovom pravcu budu u velikom zaostatku. Zbog čega im je potrebno znatno više vremena za razvoj, a samim tim i više novca. Velike kompanije su veoma brzo uvidele prednost ovog pristupa i brzo se priključile novom trendu za šta su nagrađene velikim uspehom. Sa druge strane, manje kompanije ostale su skeptične i inicijalno nisu bile spremne za ovaj veliki korak. Međutim, ubrzo su i one uvidele prednost ovog pristupa. Velikom brzinom, za samo nekoliko godina, progresivne web aplikacije postaju prvi izbor u pravljenju web i mobilnih rešenja, predstavljajući nezaobilaznu opciju za razmatranje čak i za manje kompanije koje žele da budu u koraku za velikim kompanijama [2].

2.3. Dizajniranje za mobilne uređaje kao prioritet

Era mobilnih telefona je uveliko došla, protok internet saobraćaja na mobilnim telefonima je iz godine u godinu sve veći. Iz tog razloga, dizajn mobilnih aplikacija postaje jedna od najbitnijih stavki. Prvenstveno se obraća pažnja na dizajn aplikacija za najmanje ekrane, a zatim uz određene dorade dizajn se prilagođava velikim ekranima. Neki statistički podaci ukazuju na to da preko dve milijarde ljudi danas koristi internet isključivo preko mobilnog telefona, a očekuje se da do 2025. godine taj broj skoči za 72.5% [3].

Pored navedenih prednosti ovog pristupa, takođe je bitno napomenuti Google-ov algoritam za otkrivanje aplikacija, gde je u interesu samih kompanija da naprave što bolje mobilno rešenje jer algoritam favorizuje web aplikacije prilagođene mobilnim uređajima. Takođe, bitna činjenica je da u današnje vreme, reklamiranje i oglašavanje sve više prelazi sa TV uređaja na društvene mreže, gde se korisniku plasiraju informacije na najbrži i najpregledniji način. Dizajn savremenih mobilnih aplikacija mora da obezbedi dobru brzinu i preglednost sadržaja.

2.4. Prednosti i karakteristike

Progresivne web aplikacije bez poteškoća mogu da se prikažu na praktično svim uređajima koji se danas koriste, nezavisno od veličine ekrana. Često se pri pominjanju progresivnih web aplikacije čuje i naziv aplikacije koje rade na svim uređajima, što je veoma značajno za korisnike koji često menjaju uređaje u toku rada. Omogućavaju veoma dobro korisničko iskustvo, približno nativnim aplikacijama, jer su progresivne mobilne aplikacije prvenstveno dizajnirane za mobilne uređaje. Progresivna web aplikacija radi u uslovima gde je internet u delimičnom ili potpunom prekidu, ovo je jedna od ključni osobina ovih aplikacija.

Notifikacije omogućavaju progresivnim web aplikacijama da ostanu u stalnom kontaktu sa krajnjim korisnicima kroz niz notifikacija i strategiskih poruka čiji je esencijalni cilj da korisnika drži zainteresovanim i motivisanim za ponovno korišćenje aplikacije. Optimizacija za pretraživače omogućava da progresivna web aplikacija bude lako identifikovana i vidljiva od strane pretraživača poput Google-a, Bing-a i drugih. Deljenje aplikacije je jednostavno i lako uz pomoć URL-a aplikacije, a samim tim ona je lako čitljiva od strane pretraživača. Progresivna web aplikacija omogućava da jedna aplikacija pokrije praktično sve oblasti rada i da predstavlja nativnu i web aplikaciju u jednoj što u značajnoj meri štedi finansijska i vremenska sredstva [4].

2.5. Brzina progresivnih web aplikacija

Statistički podaci ukazuju da ukoliko učitavanje stranice traje duže od 3 sekunde, postoji velika verovatnoća da će korisnik zatvoriti aplikaciju. Stopa konverzije predstavlja statistički podatak koji utvrđuje broj korisnika koji su izvršili određene akcije na aplikaciji. Viša stopa konverzije obično znači da je korisnički interfejs bolje prilagođen krajnjim korisnicima. Stopa konverzije je direktno povezana sa brzinom učitavanja stranice.

Stopa konverzije se računa po formuli *broj sesija završenih akcijom / ukupan broj sesija*. Neki statistički podaci ukazuju da je mnogo manja stopa konverzije na mobilnim uređajima nego na desktop računarima. Što znači da se jako veliki broj korisnika ne odluči da odradi kompletan proces poput kupovine, ispunjavanja formi na telefonu, već pribegava tradicionalnim desktop aplikacijama. Posledice i razlozi zašto se ovo dešava mogu biti brojni, a neki od njih su loš korisnički interfejs, nejasno prikazani podaci, komplikovan proces i drugi. Progresivne web aplikacije u značajnoj meri utiču na ove nedostatke što utiče na povećanje stope konverzije [5].

2.6. Odnos nativnih, web i progresivnih web aplikacija

Izbor tipa aplikacije koja će se praviti prvenstveno zavisi od vizije ali i od biznisa za koji je aplikacija namenjena. Danas su poznate web, nativne i progresivne web aplikacije, svaka od njih ima neke svoje prednosti ali takođe ima i neke od mana. Ključan element je da su web i nativne aplikacije potpuno različite i imaju esencijalno drugačije osobine, dok progresivne web aplikacije čine kombinaciju osnovnih osobina web i nativnih aplikacija. Web aplikacije predstavljaju loš izbor ukoliko aplikacija treba da pristupa nekim od specifičnosti za mobilne uređaje, poput kamere, notifikacija i drugih. U slučaju da je potrebno web rešenje koje ne zahteva mobilnu aplikaciju, gde će pritom ono biti veoma lako dostupno i vidljivo, web aplikacije su pravi izbor.

U slučaju da aplikacija treba da bude brza i da se iskoriste sve prednosti mobilnih uređaja, pravi izbor je nativna aplikacija. Međutim, ukoliko je ipak potrebna jednostvna aplikacija, za čiju izradu se neće potrošiti ogromna količina novca, a da pritom bude omogućeno da radi na svim uređajima, pravi izbor je progresivna web aplikacija. Progresivna web aplikacija predstavlja zlatnu sredinu, te se ovom tipu aplikacija, s obzirom da obuhvataju sve esencijalne osobine, pruža velika prednost prilikom izbora.

2.7. Ograničenja

Određene hardverske i softverske platforme i kompanije nemaju ili tek razvijaju podršku za progresivne web aplikacije. Mnoge bitne karakteristike mobilnih uređaja su delimično ili potpuno ograničene za korišćenje od strane progresivnih web aplikacija. Pošto progresivne web aplikacije koriste poslednje tehnologije i principe, njihova podrška za starije uređaje i pretraživače je limitirana ili potpuno onemogućena jer oni ne pružaju podršku za rad progresivnih web aplikacija.

2.8. Tehnički aspekt

Progresivna web aplikacija može da kontroliše internet zahteve kao i keširanje podataka ovo joj omogućuju service worker-i koji su ujedno i najbitniji element progresivnih web aplikacija. Pravilno čuvanje podataka omogućava rad progresivnih web aplikacija u režimu bez internet konekcije, jer u svakom trenutku na raspolaganju imaju podatke, kao i mnogo veću brzinu aplikacija jer omogućavaju asinhroni rad aplikacije, gde oni vode računa o komunikaciji van aplikacije. Ključna tehnologija progresivnih web aplikacija su svakako service workeri, da bi oni mogli da obavljaju svoju ulogu potrebno je obezbediti sigurno radno okruženje. Progresivna web aplikacija koristi TLS protokol kao standard za razmenu podataka između dve aplikacije a on omogućava da internet stranica bude sigurna.

Takođe bitan element svake progresivne web aplikacije je manifest fajl koji predstavlja konfiguracioni fajl sa svim potrebnim informacijama koje omogućavaju da se aplikacija može prilagoditi svim platformama, odnosno omogućuje da se web aplikacija prezentuje kao nativna aplikacija ili najpribližnije nativnoj aplikaciji. Bitno je napomenuti da progresivne web aplikacije uz pomoć service worker-a osnovni deo aplikacije drže keširan i on omogućava prikazivanje aplikacije u stanju kada je internet konekcija u prekidu. Prvo se učitava kostur aplikacije, odnosno ključni elementi a zatim se dinamički učitavaju ostali podaci [6].

3. APLIKACIJA ZA RAD U BIBLIOTECI

Library je aplikacija namenjena za podršku rada biblioteke. Razvijena je kao progresivna web aplikacija, tako da može da se prikaže na web-u ali i na mobilnim uređajima. Aplikaciju mogu da koriste dva tipa korisnika: administratori (zaposleni u biblioteci) i obični korisnici (korisnici usluga biblioteke). Cilj je da se uz pomoć Ionica omogući prikazivanje web rešenja i na mobilnim uređajima, u ovom slučaju na Androidu i iOS-u.

3.1. Implementacija aplikacije

U implementaciji rešenja osnovni zadatak je da se razvije frontend aplikacija koja će da obezbedi implementaciju svih zahteva navedenih u specifikaciji aplikacije. Library aplikacija je sačinjena od niza komponenti grupisanih u module. Komponenta predstavlja gradivni element stranica i njena složenost zavisi od zahteva aplikacije. Modul predstavlja grupu komponenti, odnosno komponente grupisane po nekom kriterijumu. Obično to mogu biti neke zajedničke osobine ili namena, ali isto tako i područje njihove primene. Takođe, moduli mogu da ujedno čine i komponente koje su iskoristive od strane više drugih

modula, što ih čini univerzalnim komponentama aplikacije. Modul takođe može da objedinjuje i druge elemente aplikacije kao što su servisi, ali isto tako postoje i moduli koji vode računa o samom rutiranju unutar aplikacije.

Glavni problem nastaje kada je potrebno učitati veliki broj modula prilikom inicijalnog prikazivanja aplikacije. Što u značajnoj meri utiče na same performanse aplikacije i učitavanje postaje znatno duže. Da bi se ovo sprečilo module organizujemo po stranicama same aplikacije i sve komponente stranice svrstavamo u jedan modul. Inicijalno je potrebno učitati modul samo stranice koja se prikazuje dok ostali moduli mogu ostati ne učitani. Da bi se ovaj sistem učitavanja ostvari koristimo odloženo učitavanje (eng. lazy loading).

Pored navedenog imamo i modul za rutiranje koji vodi računa o rutiranju i inicijalno će se učitati samo početni ekran sa svim definisanim bibliotekama. modul za rutiranje takođe ima opciju da uz pomoć zaštitnika rute (eng. Guard) za svaki od modula koristeći funkciju canActivate navede da li se određenoj ruti može pristupiti sa ili bez autentifikacije. Zaštitnici ruta se mogu implementirati tako da u slučaju da se pristupi nekoj ruti koja zahteva autentifikaciju, korisnika usmeri na stranicu za logovanje, a zatim redirektuje nazad na željenu stranicu. Library aplikacija ima implementiran dodatak za učitavanje sadržaja (eng. interceptor), koji vodi računa i presreće sve zahteve poslate ka backend serveru, vodeći računa o postavljanju odgovarajućih zaglavlja (eng. header) zahteva koji su potrebni da bi server znao da se radi o autentifikovanom korisniku. Takođe, pored navedenog vodi računa o svim potencijalnim greškama koje server može da vrati i blagovremeno korisnika obaveštava o nastalom problemu. Dobar korisnički interfejs u svakom trenutku korisnika mora da obavesti o svim dešavanjima unutar aplikacije, tako da korisnik nikada ne ostane u nedefinisanoj stanju.

3.2. Integracija sa Ionicom i kreiranje PWA

Cilj aplikacije je da prikaže mogućnost da se web aplikacija bez velikih poteškoća može pretvoriti u progresivnu web aplikaciju. Integracija Ionic-a u Angular aplikaciju je veoma jednostavna, Angular CLI omogućava uvezivanje svih potrebnih biblioteka Ionica. Nakon toga je potrebno inicijalizovati postojeći projekat sa Ionicom pozivanjem *ionic init* komande, koja od korisnika zahteva unos imena projekta, a zatim izvršava inicijalizaciju. Angular aplikacija postaje ujedno i Ionic aplikacija, sa mnogo više mogućnosti a sledeći korak je konfiguracija potrebna za kreiranje progresivne web aplikacije.

Dve glavne karakteristike progresivnih web aplikacija su: service worker i web manifest. Angular omogućava instaliranje biblioteke koja ima mogućnost automatizacije njihovog dodavanja.

Pokretanjem komande *ng add @angular/pwa* biće modifikovano nekoliko fajlova i biće dodata dva nova fajla: *ngsw-config.json* i *manifest.webmanifest*. Fajl *ngsw-config.json* predstavlja konfiguracioni JSON fajl service workera dok fajl *manifest.webmanifest* predstavlja konfiguracioni JSON fajl koji se koristi da odredi izgled aplikacije koja se prikazuje.

Nakon završenog lokalnog testiranja, poslednja verzija aplikacija je postavljena (eng. Deploy) na Firebase server. Kao prvi korak potrebno je napraviti nalog na Firebase-u.

Zatim je potrebno izvršiti autentifikaciju u terminalu pokretanjem komande *firebase login*, a potom pokretanjem *firebase init* komande potrebno je napraviti novi projekat i dati mu ime. Po inicijalizaciji Firebase projekta, potrebno je izvršiti dodatnu produkcijsku konfiguraciju nakon čega se vrši postavljanje aplikacije na Firebase server. Po uspešnom objavljivanju aplikacije ona postaje dostupna na odabranom domenu. Sa URL-om aplikacije uz pomoć Lighthouse alata omogućeno je analiziranje aplikacije na prosluđenoj putanji. Po završenom analiziranju otvara se panel u kome je prikazana statistika sajta, kao i sugestije kako da se svaki segment unapredi [7].

Takođe, s obzirom da se radi o progresivnoj web aplikaciji, ona omogućava da se na mobilnim telefonima sačuva prečica na početnom ekranu, tako da joj se kasnije može pristupiti kao nativnoj aplikaciji. Dodavanjem aplikacije na početni ekran, ona će postati lako dostupna, poput aplikacija koje su instalirane uz pomoć prodavnice aplikacija. Nakon toga, u bilo kom trenutku je omogućen pristup aplikaciji bez kucanja adrese u internet pretraživaču.

4. ZAKLJUČAK

Povećano korišćenje mobilnih telefona kao posledicu ima potrebu za sve većim brojem mobilnih aplikacija. Zbog prednosti koje mobilne aplikacije pružaju postale su nezaobilazan i ključan faktor u komunikaciji sa krajnjim korisnikom. Ova promena na tržištu dovela je do toga da web aplikacije više nisu dovoljne, te da svaka od njih zahteva i svoje mobilno rešenje. Pravljenje više od jedne aplikacije zahteva više vremena, više finansijskih sredstava, a ujedno i adekvatno obučene programere za ovaj poduhvat. Nastale promene podstakle su programere da razmisle o jedno-aplikacijskom rešenju, jednoj aplikaciji koja će predstavljati i mobilno i web rešenje. U tom svetlu pojava progresivnih web aplikacija usledila je kao uzročna posledična potreba za rešenjem ovog problema.

Predstavljena je velika potreba za što boljim korisničkim interfejsom, sa ciljem poboljšanja korisničkog iskustva. Dizajn mobilnih aplikacija postao je ključan element u svetu web-a, gde je veliki teret pao na leđa dizajnera koji su postali primorani da se prilagode najmanjim ekranima, a da pritom pruže najbolje moguće korisničko iskustvo, uz održavanje jednostavnosti i efikasnosti aplikacije. Nedostaci koji danas postoje kod progresivnih web aplikacija predstavljaju ključnu tačku za razvoj boljeg i optimalnijeg puta za razvoj nekih budućih jezika i radnih okvira. Ključan element u razvoju budućih aplikacija uvek predstavlja korisničko iskustvo i nedostaci postojećih rešenja.

5. LITERATURA

- [1] <https://www.divante.com/pwabook/chapter/02-the-history-of-pwas>, Divante eCommerce Software House, “*The history of PWA development*” (pristupljeno u martu 2023.)
- [2] <https://medium.com/progressivewebapps/history-of-progressive-web-apps-4c912533a531>, ScandiPWA, “*History of progressive web applications*” (pristupljeno u martu 2023.)
- [3] <https://xd.adobe.com/ideas/process/ui-design/what-is-mobile-first-design>, Justin Morales, “*Mobile First Design Strategy: The When, Why and How*” (pristupljeno u martu 2023.)
- [4] <https://inoxoft.com/blog/benefits-of-progressive-web-apps-pwa-advantages-and-disadvantages>, Nazar Kwartalni, “*Benefits of PWA*” (pristupljeno u martu 2023.)
- [5] <https://www.divante.com/pwabook/chapter/01-introduction-to-pwa-technology>, Divante eCommerce Software House, “*Introduction to Progressive Web Apps*” (pristupljeno u martu 2023.)
- [6] <https://medium.com/@tysonpaul89/what-i-learned-about-progressive-web-app-pwa-part-2-4074137301a7>, Tyson Paul, “*Progressive Web App (PWA)*” (pristupljeno u martu 2023.)
- [7] <https://ionicthemes.com/tutorials/the-complete-guide-to-progressive-web-apps-with-ionic4>, Agustin Haller, “*The Complete Guide To Progressive Web Apps with Ionic Framework*” (pristupljeno u martu 2023.)

Kratka biografija:



Dejan Predojević rođen je 16.06.1996. godine u Vrbasu. Završio je Srednju tehničku školu „Mihajlo Pupin” u Kuli 2015. godine. Fakultet tehničkih nauka u Novom Sadu je upisao školske 2015/2016. godine. Osnovne akademske studije završio je 2019. godine. Iste godine upisao je master akademske studije, modul Softversko inženjerstvo. Kontakt: dejanpredojevic11@gmail.com