

# Najpopularnije JavaScript biblioteke

Đorđe Jovanović

**Sadržaj** —Popularnost JavaScript jezika je dovela do pojave velikog broja biblioteka i tehnologija. Zbog velikog izbora nije uvek lako pronaći onu biblioteku koja je vredna ulaganja energije i truda. Većina njih je otvoreni kod, što je jedan od glavnih razloga popularnosti, ali je i razlog za pojavu loših, nezavršenih biblioteka punih grešaka. U radu su prikazane statistike biblioteka na najvećim repozitorijumima, kao i rezultati anketa. U radusu takođeupoređene trenutno najbolje biblioteke, izabrane na osnovu tih rezultata i statistka, zajedno sa njihovim opisom i za šta ih je najbolje koristiti.

**Ključne reči** —Anketa o popularnosti biblioteka,Front-end biblioteka,JavaScript biblioteka,Otvoreni kod

## I. UVOD

**O**VA rad ima za cilj da pomogne svakome ko želi više da se upozna sa JavaScript jezikom i najnovijim tehnologijama, pre svega u razvoju veb aplikacija. Takođe, ovaj rad može pomoći u odabiru prave biblioteke u zavisnosti od potreba. U delu II opisanisu najkorišćeniji frejmvorci, jedan po jedan, ne ulazeći u detalje, već predstavljajući osnovne prednosti i mane, osnovna svojstva, kao i situacije u kojima ih je najbolje koristiti. U delu III su predstavljeni njihovi rejtinzii rezultati velike ankete sprovedene tokom prošle godine, i upoređeni sa rezultatima ispitivanja sprovedenim u Srbiji.

## II. BIBLIOTEKE

### A. Angular

Angular [1]svoju popularnostnajviše duguje tome što je podržan od strane Google-a.Iako jeobjavljen 2009. godine i dalje je među najboljima što se tiče front-endfrejmvorka. Druge prednosti su i to što je produkcija koda brza i

Đorđe Jovanović, Računarski fakultet, Univerzitet Union, Knez Mihailova 6/6, Beograd, [djolejov@gmail.com](mailto:djolejov@gmail.com)

svaki deo aplikacije se može lako testirati. Interesantno je pomenuti da podržava MVW paradigmu (Modal-View-Whatever) gde se pod „whatever“ misli na bilo šta što je potrebno (Controller, ViewModel, itd). Ovaj frejmwork koristi dvosmerno povezivanje podataka (two-way data binding) kao glavni koncept. Interakcija korisnika sa interfejsom ažurira pogled, što zatim ažurira i model, i sinhronizacija je uvek obezbeđena. Angular daje veliku fleksibilnost za razdvajanje prezentacione i biznis logike. Najviše se koristi za aplikacije sa jednom stranom (Single-Page Applications) i koriste ga startup kompanije i kompanije srednje veličine. Međutim, ima i mana. Njegova kompleksnost zahteva veće znanje i teži je za učenje. Indeksiranje od strane pretraživača je otežano jer se koristi jedna strana, a oni pristupaju samo statičnim stranama. Angular 2.0 je objavljen nedavno, u oktobru 2016, i iako nije mnogo korišćen za sada, zbog velikog interesovanja očekuje se da i on bude popularan. U članku [2] tvrde da su performanse Angular 2.0 bolje na mobilnim i desktop uređajima. Dodaje mnoge karakteristike kao što su veb radnici i serversko iscrtavanje. Preporučuje se svima koji žele snažan MVC frejmwork sa snažnom i modernom infrastrukturom, koja se brine o svemu za pravljenje aplikacije sa jednom stranom. Najavljen je i Angular 4 za mart 2017.

### *B. React*

Ovu biblioteku [3] je napravio Facebook i koristi se na njemu, Instragramu i mnogim drugim poznatim sajtovima. Glavna uloga mu je pravljenje interfejsa i prikazivanje i iscrtavanje podataka na stranici. U MVC paradigmi za React se može reći da je View, iako nudi neke dodatne stvari kao što su komponente sa stanjima i mapiranje ulaza na promenu stanja. Za razliku od Angulara, ova biblioteka se bazira na jednosmernom protoku podataka. Uprošteno postoji sledeći ciklus. React snabdeva komponente ulazom preko „props-a“ i uslovno iscrtava DOM ukoliko se određeni deo promenio. Promene podataka tokom ove faze ne mogu ponovo pokrenuti iscrtavanje sve do sledeće faze iscrtavanja. Nakon iscrtavanja sledi faza događaja – događaji se oslušuju i delegiraju jednom oslušivaču događaja u korenu DOM stabla. Događaji se tu hvataju i tu se podaci mogu menjati. Bilo kakva promena podataka povlači njihovo iscrtavanje u sledećem koraku. Ova biblioteka je promenila način na koji ljudi razmišljaju o arhitekturi front-end-a uzimajući ideje iz funkcionalnog programiranja. Zajednica koja podržava ovaj frejmwork je sve veća, što pokazuje koliko ljudi doprinosi njegovom razvoju na javnim repozitorijumima. Samim tim je i brzina kojom se razvija velika. React komponente se mogu vrlo lako ponovo iskoristiti ili čak staviti u javnu upotrebu. Virtualni DOM poboljšava performanse. Takođe se i vrlo lako debuguje. Neke od mana su nedostatak dokumentacije, kao i to da

jepočeticima u web razvoju donekle teško da ga savladaju, ali najveća je ta što ovaj frejmwork nije dovoljan sam po sebi, te programeru može biti veoma teško da izabere dodatne biblioteke. Svakako se izuzetno preporučuje za kompleksne i skalabilne interfejs, i odabirom ove biblioteke skoro je nemoguće pogrešiti.

### *C. Vue*

Odlike ovog frejmworka [4] su jednostavnost i brzina. Koristi MVVM (Model-view-viewmodel) dizajn paradigmu umesto tipične MVC paradigme. Pruža dvosmerno povezivanje podataka kao i Angular, serversko isrcrtavanje kao Angular 2 i React, i još mnogo toga. Najviše se fokusira na View sloj, pa programer mora sam da se pobrine za ostale komponente, ili da iskoristi neki drugi frejmwork za njih. Najpogodniji je za brz razvoj aplikacija. Skalira se kako projekat raste, pa može postati i krajnje rešenje kompanijama koje performanse stavljaju ispred dobro organizovanog i struktuiranog koda. Daje mnogo više slobode nego Angular, ali manje od Backbone-a za koji možemo reći da je tabula rasa. Vue pravi balans između onoga što nudi i onoga što ostavlja programeru da sam uradi. Zbog svog minimalizma lako se uči u poređenju sa ostalim opcijama, a pritom daje vrlo moćna svojstva drugih biblioteka.

### *D. Backbone*

Objavljen 2010. godine, cilj ovog frejmworka [5] bio je minimalizam, i takav je do danas ostao. To je vrlo verovatno razlog zašto pripada najboljim front-end bibliotekama. Podržava MVC dizajn paradigmu. Daje web aplikaciji strukturu pružajući modele sa ključ-vrednost povezivanjem i događajima vezanim za model, daje kolekcije sa korisnim funkcijama i apstrahuje DOM u pogledu, vezujući ih događajima. Svojstva vredna pomena su i rutiranje, podrška za RESTful API, kao i upravljanje stanjima. Filozofija minimalizma je pokušaj da se pronađe najmanji mogući skup struktura podataka i primitiva interfejsa koji su korisni i dovoljni za pravljenje web aplikacije sa JavaScript-om. Autor je takođe napravio Underscore, kao i CoffeScript, tako da se njegovom radu može verovati. Biblioteka zavisi od JQuery-a i Underscore-a. Veličina biblioteke je mala. Vrlo lako se uči. Izuzetno je fleksibilna i može se koristiti kao baza za razvoj bilo čega. Performanse su dobre zbog jednostavne strukture. Zajednica koja se formirala oko Backbone frejmworka vrlo je aktivna i velika. Neke od mana su nedostatak dvosmernog povezivanja podataka, teže testiranje, sporo ažuriranje same biblioteke, kao i nedostatak mnogih svojstava uključenih u veće biblioteke, što samim tim povlači da programer mora imati veće znanje

kako bi radio sa dodatnim potrebnim bibliotekama koje to treba da nadoknade. U svakom slučaju, mnoge velike kompanije koriste Backbone, kao što su Airbnb, Pinterest, Reddit, WordPress.com i preporučuje se svakome ko želi jednostavnost.

#### *E. Ember*

Kao kombinacija Angular i React biblioteke, ovaj frejmwork [6] je bio proglašen najboljim 2015. ispred njih. Objavljen je 2011. i od tada je stvorio ogromnu zajednicu. Redovna ažuriranja, mnogi priručnici i vodiči rezultat su te zajednice. Možete očekivati i brzu pomoć za bilo koji problem na koji naiđete tokom razvoja web aplikacije sa Ember frejmworkom. Kao i Angular, ima dvosmerno povezivanje podataka, pa su pogled i model uvek sinhronizovani. Zasniva se na MVVMdizajn paradigmi. Što se tiče sličnosti sa React-om, Ember pruža serverski virtuelni DOM, i povećava performanse i skalabilnost. U paketu dolazi i njegov sopstveni mehanizam za šablon (templating engine) koji se automatski ažurira. Podržava i rutiranje. Međutim, sve to ide na uštrb fleksibilnosti i kontrole koju programer ima. Ima sopstvena pravila koja programer mora da isprati. Jednostavan je za učenje, ima stabilne performanse. Zbog svih ovih pogodnosti, veličina ove biblioteke je velika. Strukture koje su na visokom nivou mogu dovesti do mnogih grešaka. Moguće je pronaći i zastarele priručnike i vodiče na internetu. Sve u svemu, mnogo poznatih kompanija koristi ovaj frejmwork kao što su Yahoo, KickStarter, LinkedIn, Zendesk, GROUPON. Najčešće se koristi za kompleksne veb aplikacije.

#### *F. Meteor*

Meteor [7] je full-stack frejmwork, što znači da vam je samo on potreban za razvoj celokupnog sajta ili aplikacije, od kraja do kraja. Pokriva sve faze softverskog razvojnog ciklusa i može se koristiti za mnoštvo platformi – desktop, mobilne i web aplikacije. Programer sve radi u JavaScript-u. Sadrži server baziran na Node.js-u, front-end biblioteke i command line alat. Umnogome olakšava posao time što se brine o mnogim dosadnim procesima kao što su vezivanje, minimizovanje i spajanje fajlova, itd. Ima veliku zajednicu, te samim tim i dobru podršku, što je vrlo pozitivna stvar jer nije lak za učenje i programeru je potrebnu vreme da se dobro upozna sa njim. Performanse su odlične iz razloga što nema komuniciranje između različitih programskih jezika na različitim nivoima, i promene u bazi podataka se istog trenutka pojavljuju na interfejsu i obrnuto. Kompanije kao što su Mazda, IKEA, Honeywell koriste ga za razvoj svojih aplikacija u realnom vremenu. Vrlo je moćan i dobri programeri mogu napraviti skoro bilo šta sa ovom

platformom, i odličan je izbor za razvoj veb aplikacija od nule.

### *G. Redux*

Upravljanje stanjima je relativno nova kategorija u JavaScript-u. U toj kategoriji, vrlo je jasno da je Redux [8] ubedljivo najpopularniji. Pruža nam transakciono, determinističko upravljanje stanjem aplikacije. Čak i ako ga programer nikad ne bude koristio, vredan je učenja zbog načina na koji bi razmišljao o reduktorima (reducers), koji su funkcije opšte namene za prolazak kroz kolekcije i izvlačenje vrednosti iz njih, i gde i kako ih može koristiti. Zbog toga mnogi smatraju da je ova biblioteka obavezna veština koju svaki JavaScript programer treba da poseduje.

### *H. Mocha*

Mocha[9] je JavaScript frejmwork za testiranje. Pokreće se u Node.js i u pregledaču. Olakšava asinhrono testiranje sa „Done“ callback funkcijom. Nezavistan je od odabira biblioteke za potvrdu (assertion library). Dobro je integrisan sa Node.js-om, pa se preporučuje za aplikacije koje su i same bazirane na Node.js-u. Ima veliki izbor ugrađenih izveštaja, a moguće je napraviti i sopstvene. Izveštaji su u boji, pa je pregled uspešnih i neuspešnih testova odličan. Mana je ta što se može činiti komplikovanim za početnika. Takođe, svi testovi se izvršavaju u jednom procesu, što podrazumeva deljenu memoriju bez nezavisnosti i izolacije.

### *I. Node*

Mnogi su već čuli za Node.js [10] i istina je da se već svuda pominje, ali je ova biblioteka danas zaista neizostavna bilo gde gde se pominje JavaScript. Node beleži najveći rast u svakom pogledu i pruža mnogo više od običnog okruženja za razvijanje. Uz pomoć NPM-a moguće je upravljanje lokalnim paketima direktno iz komandne linije. Pun je korisnih alata, modula i programeri širom sveta doprinose tom razvojnom okruženju. Napravljen je na Chrome V8 JavaScript engine-u, koristi neblokirajući model vođen događajima. Primarna svrha mu je pravljenje sajtova za zajednice, sajtovima sa striming sadržajem, kompleksnim sajtovima sa jednom stranom, i drugim interaktivnim aplikacijama. Moglo bi se reći da je njegovo poznavanje obavezno za svakoga ko se bavi JavaScript-om.

Na tabeli 1 je prikazan uporedni pregled nekih svojstava biblioteka koje služe za razvoj front-enda i koje su predstavljene u ovom radu.

TABELA 1: PREGLED SVOJSTAVA.

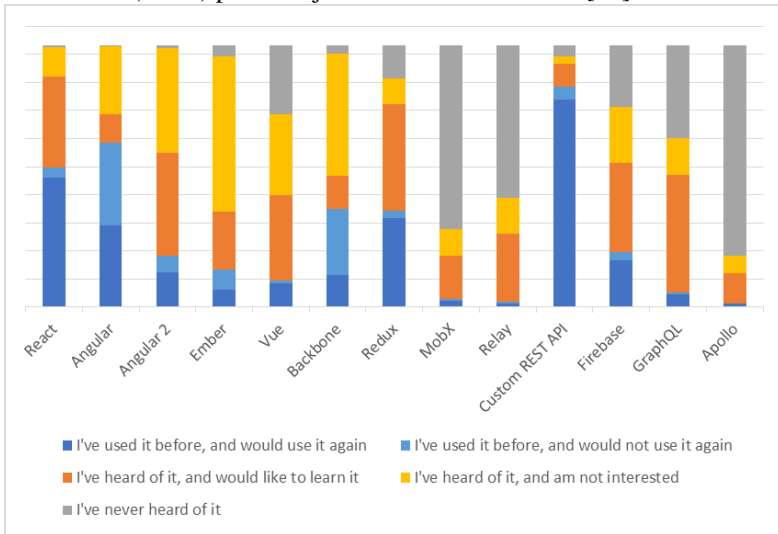
<i>Frejmvork</i>	<i>Dvosmerno povezivanje podataka</i>	<i>Lako se uči</i>	<i>Mala veličina biblioteke</i>	<i>Podržava MVC(ili varijaciju)</i>	<i>Podrđava rutiranje</i>
Angular	✓		✓	✓	✓
React					
Vue	✓	✓	✓	✓	✓
Backbone		✓	✓	✓	✓
Ember	✓	✓		✓	✓
Meteor					✓

### III. STATISTIKE

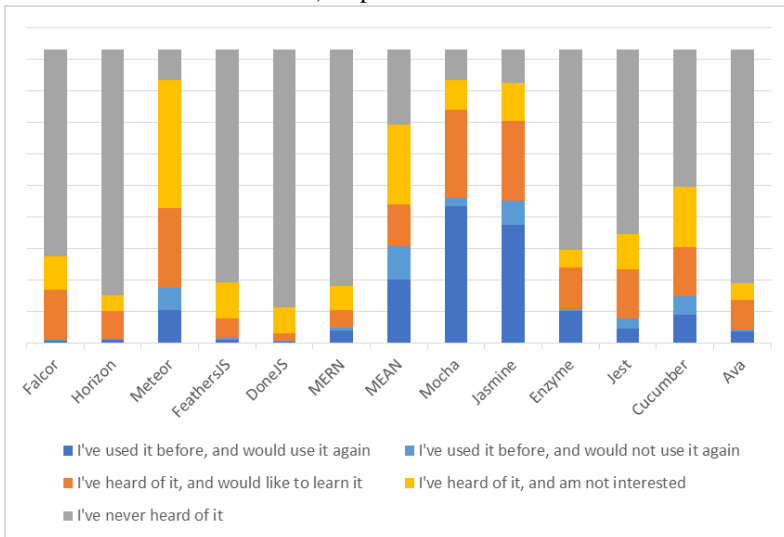
Izbor odgovarajuće JavaScript biblioteke mogli bismo da napravimo i u zavisnosti od izbora drugih ljudi, i mišljenja onih koji su te biblioteke već isprobali. Sprovedeno je mnogo anketa na kojima su ljudi ispitivani o njihovim budućim interesima, o tome da li su uopšte čuli za određene biblioteke, i da li bi uopšte želeli da ih nauče. Ispitivana su i mišljenja o alatima koje su već koristili, da li bi ih ponovo koristili, i da li bi ih preporučili nekom drugom. Neke od anketa vredne pomena nalaze seu[11] i [12]. Još jedan način da se odredi popularnost jeste na osnovu broja doprinosa ili na osnovu rejtinga na repozitorijumima.

Kada je u pitanju JavaScript flavor, ES6 je novi standard. Ogromna većina onih koji ga već ne koriste želi da ga nauči. CoffeScript je uveliko izgubio interesovanje, i ne samo da su ljudi prestali da ga koriste, već ni oni koji su čuli za njega nemaju želje da ga nauče. Što se tiče rešenja za front-end biblioteke, u samom vrhu su React i Angular. Iako je veći broj ljudi koristio Angular, mnogo više njih je reklo da bi nastavili da koriste React u budućnosti. Angular 2 i Vue su noviteti i postoji interesovanje za njih od strane onih koji još uvek nisu imali prilike da ih koriste, ali bitno je uzeti u obzir da je manje ljudi čulo za Vue. Biblioteke za upravljanje stanjima su drastično manje poznate, osim Redux-a. Ljudi su vrlo zadovoljni tom bibliotekom, i velika većina onih koji su čuli imaju interesovanje i da je nauče, pa slobodno možemo reći da se ne može pogrešiti sa njom. Zamene za REST API kao standarda još uvek nema, ali Firebase i GraphQL su vredni pomena i možda u budućnosti budu mogli da se probiju. Full-Stack frejmvorci su dosta slabije poznati i mnogi bi rekli da ovaj koncept jednostavno ne pije vodu. Dva izuzetka su Meteor i MEAN gde je više ljudi

koristilo MEAN, a više ljudi je čulo za Meteor. Mocha i Jasmin, biblioteke za testiranje, izjednačene su u svemu, interesovanju i broju ljudi koji ih koristi. Na slikama 1 a) i 1 b) prikazan je deo rezultata ankete iz [11].



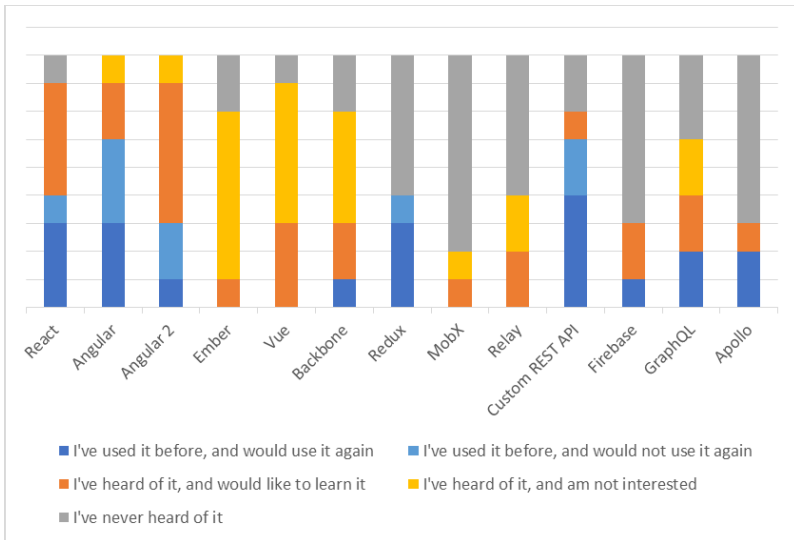
Sl. 1. a) Popularnost biblioteka



Sl. 1. b) Popularnost biblioteka

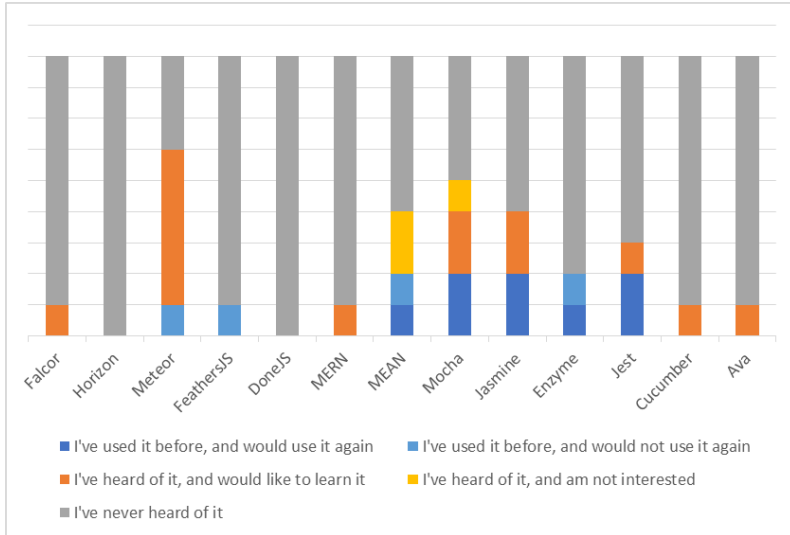
Takođe, bilo bi vrlo interesantno uporediti ove rezultate sa mišljenjem u Srbiji. Na osnovu sprovedene ankete na kojoj su postavljena ista pitanja, kao i na osnovu pojedinačnih razgovora sa programerima koji se bave veb aplikacijama možemo izvesti sledeće zaključke.

JavaScript programeri u Srbiji u stopu prate trendove kada su u pitanju front-end biblioteke. React i Angular su najpopularniji, i postoji slično interesovanje za Angular 2, Vue, Ember i Backbone. Međutim, ono što je vrlo zanimljivo je da je svest o ostalim vrstama frejmvorka, za testiranje, upravljanje stanjima i full-stack frejmvorcima, dosta manja. Većina nije čula za njih, a vrlo mali broj je koristio bilo koji. Redux ili koriste, ili uopšte nisu čuli za njega, što je nešto što bi definitivno trebalo da se promeni. Na slikama 2 a) i 2 b) prikazani su grafici sa rezultatima.



Sl. 2. a) Popularnost biblioteka u Srbiji





Sl. 2. b) Popularnost biblioteka u Srbiji

Još jedan pokazatelj da je neki frejmwork vredan pažnje, jeste broj zvezdica na GitHub repozitorijumu. Korisnici označavaju zvezdicu kada im se nešto čini interesantnim, žele da pristupe tom projektu kasnije, ili kada jednostavno žele da pokažu zahvalnost. Na tabeli 2 prikazani su ovi podaci.

TABELA 2: GITHUB STATISTIKA.

<i>Frejmwork</i>	<i>Zvezdice</i>	<i>Forks</i>
React	58240	10573
Angular	54456	27073
Vue	40741	5033
Meteor	36534	4555
Node	30919	5678
ExpressJS	29610	5421
Redux	27512	4976
Backbone	25976	5640
Ember	17444	3646

Broj račvanja(forks) može nam reći nešto o doprinosu ljudi biblioteci. Veći broj može da znači da je više ljudi učestvovalo u razvijanju. Može značiti da projekat ima puno ažuriranja i ispravki bagova, ili mnogo dodatih, novih funkcija. U svakom slučaju sigurno je pokazatelj interesa programera.

#### IV. ZAKLJUČAK

Neki će reći da JavaScript ide u dobrom smeru, da postaje pouzdan i moderan jezik. Neki će reći da vlada haos usled prevelikog broja međusobno zavisnih biblioteka, prepunih nedovršenih stvari i grešaka. Iz tog razloga nam je lakše da posmatramo statistiku, pre nego pojedinačna mišljenja. Neki od alata opisanih u ovom radu moždaće ubrzo prestati da se koriste, i tada će ih zameniti neki novi, bolji. Rezultati anketa predstavljaju samo trenutno mnjenje i jedino možemo biti sigurni u brz rast i velike promene JavaScript jezika.

#### LITERATURA

- [1] <https://angularjs.org/>
- [2] Angular University „AngularJs vs Angular 2 - An In-Depth Comparison“ Available: <http://blog.angular-university.io/introduction-to-angular2-the-main-goals/>
- [3] <https://facebook.github.io/react/>
- [4] <https://vuejs.org/>
- [5] <http://backbonejs.org/>
- [6] <http://emberjs.com/>
- [7] <https://www.meteor.com/>
- [8] <http://redux.js.org/>
- [9] <https://mochajs.org/>
- [10] <https://nodejs.org/en/>
- [11] Sacha Greif “The state of JavaScript 2016” Available: <http://stateofjs.com/>
- [12] Ashley Nolan “The State of Front-End Tooling 2016” Available: <https://ashleynolan.co.uk/>

#### ABSTRACT

The number of JavaScript libraries and technologies has risen due to the popularity of JavaScript language. It is not always easy to pick the right library that is worth the energy and effort. Most of them are open source, and that is the reason for their popularity, but also the reason for appearance of bad, unfinished libraries full of bugs. In this paper the libraries repository statistics and survey results are shown. The best libraries, chosen by those results and statistics, are also compared, and their description is presented.

### **THE MOST POPULAR JAVASCRIPT LIBRARIES**

Đorđe Jovanović