**FAKULTET ZA INFORMACIONE TEHNOLOGIJE**

**PODGORICA**

**DOKUMENTACIJA PROJEKTA**

**ZA OSNOVNE STUDIJE**

**OPTIMIZACIJA REUSABLE KODA U VUE.JS FRAMEWORKU**

**Profesor:**

**Prof. dr Snežana Šćepanović**

**Student:**

**Borko Smolović 46/17**

**Podgorica, JUL 2020**

**Sadržaj**

[1. UVOD 1](#_Toc46193427)

[2. Vue.js 2](#_Toc46193428)

[3. Vuetify 5](#_Toc46193429)

[4. Problemi i implementacija OPTIMIZACIJE REUSABLE KODA U VUE.JS FRAMEWORKU 8](#_Toc46193430)

[5. Zaključak 9](#_Toc46193431)

[6. LITERATURA 10](#_Toc46193432)

1. UVOD

Postoje brojni direktni i praktični razlozi za implementaciju optimizacije reusable koda u svim vrstama programiranja. Ona nosi brojne benefite kao sto su jednostavnost, preglednost, razumljivost i lakoća budućih izmjena koda.

Optimizacija se direktno odnosi na pravljenje univerzalnih komponenti kao i metoda koje mogu da se iskoriste na više mjesta, cime se olakšava rad sistema i ubrzavaju potencijalne izmjene, jer ukoliko se komponenta mijenja na jednom mjestu izmjene se primijene svudje gdje se komponenta koristi, umjesto da pišemo posebne komponente ili metode za svaki slučaj korištenja čime se dobija višak i ponavljanje koda, kao i povećanje kompleksnosti izmjena jer se izmjene moraju posebno implementirati na zasebnim komponentama.

Glavni benefiti se primjećuju kod velikih projekata koji se mogu sastojati od više stotina pa i hiljada komponenti, medju kojima se dosta stvari ponavlja.

Grupisanjem ovih komponenti u slične i pravljenjem univerzalnih dobija se ogromna ušteda kako na vremenu tako i na ostalim potrebnim resursima za realizaciju projekta, i upravo sa veličinom projekta skaliraju i benefiti korištenja ovog tipa programiranja.

NEKE OD GLAVNIH KORISTI I PREDNOSTI OPTIMIZACIJE KODA:

• Povećanje efektivnosti i efikasnosti web aplikacija

• Razvijanje preglednih, razumljivih i lakih za izmjenu web aplikacija

• Smanjenje viška koda

• Optimizacija resursa potrebnih za izradu web aplikacija

• Značajne uštede na vremenu izrade web aplikacija

• Poboljšanje performansi i održivosti sistema

2. Vue.js

U prethodnih 10 godina web stranice su postale mnogo naprednije i dinamičnije zahvaljujući javascript-u.

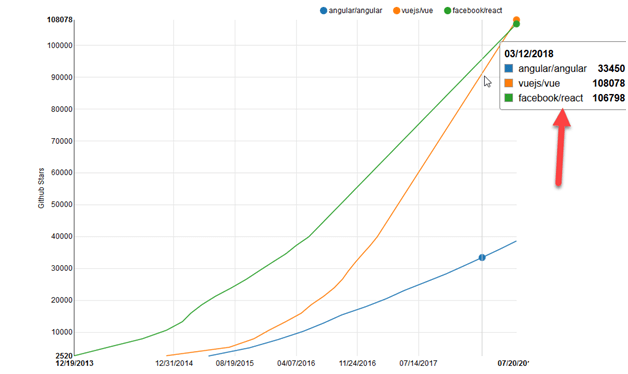
Ključne izmjene predstavljaju pomjeranje ogromne količine koda sa servera u naše pretraživače, ostavljajući nas sa javascript kodom koji je povezan sa mnoštvom html-a i css-a ali sa slabom organizacijom. Ovaj problem je direktno povezan sa nastankom javascript framework-a kao sto su vue i angular, react i slično.



Slika 2.1 React vs Vue vs Angular

Ovaj projekat je rađen u VueJS framework-u. Od gore navedenih framework-a VueJS je najmladji, objavljen 2014 godine, što je veoma mlado u svijetu programskih jezika (za referencu Java je nastala 1995 godine), ali je ujedno i najoptimizovaniji, i ima najbolju podršku od njih, redovne update, bug fix-ove, kao i najaktivniji community gdje svi mogu da učestvuju u razvoju kao i u predlozima za nadogradnje, izmjene i dodatke.

Upravo ovakav pristup i fokus na pristupačnost, raznovrsnost, performanse, jednostavnost i testabilnost su privukli ogroman broj kako senior developera tako i onih kojima je kodiranje samo hobi i postali jedan od najprepoznatljivijih frameworka za kratak period.



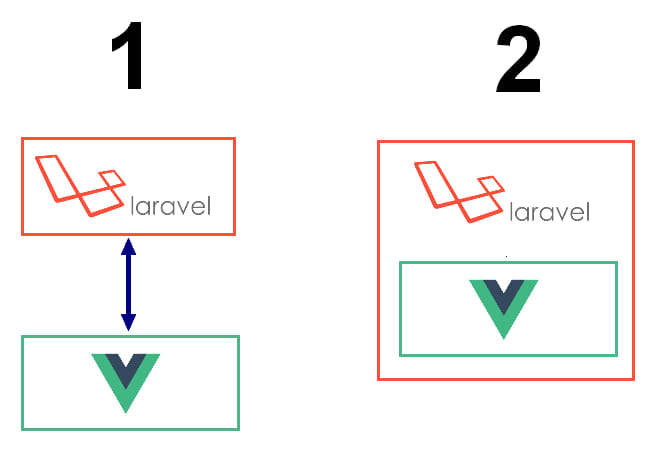
Slika 2.2 Popularnost VueJS, Angular i React frameworka

Vue je progresivan framework što znaci da može da se implementira u samo jedan dio web aplikacije, i da radi zajedno i paralelno sa drugim frameworkom. Česta kombinacija i ujedno jedna od najjačih su VueJS i Laravel.

Izdvajamo dva tipa korištenja VueJS + Laravel kombinacije.

* Prvi tip predstavlja posebne projekte koji su povezani ali mogu da funkcionišu nezavisno jedan od drugog.
* Dok drugi tip predstavlja implementaciju VueJS u Laravel projektu kao dio njega pri čemu vue zavisi od laravela.

Obično se kod većih projekata VueJS i Laravel razdvajaju i komuniciraju pomoću dodatnog Vue-ovog alata zvanog Core, Vuex i Vue-Router.



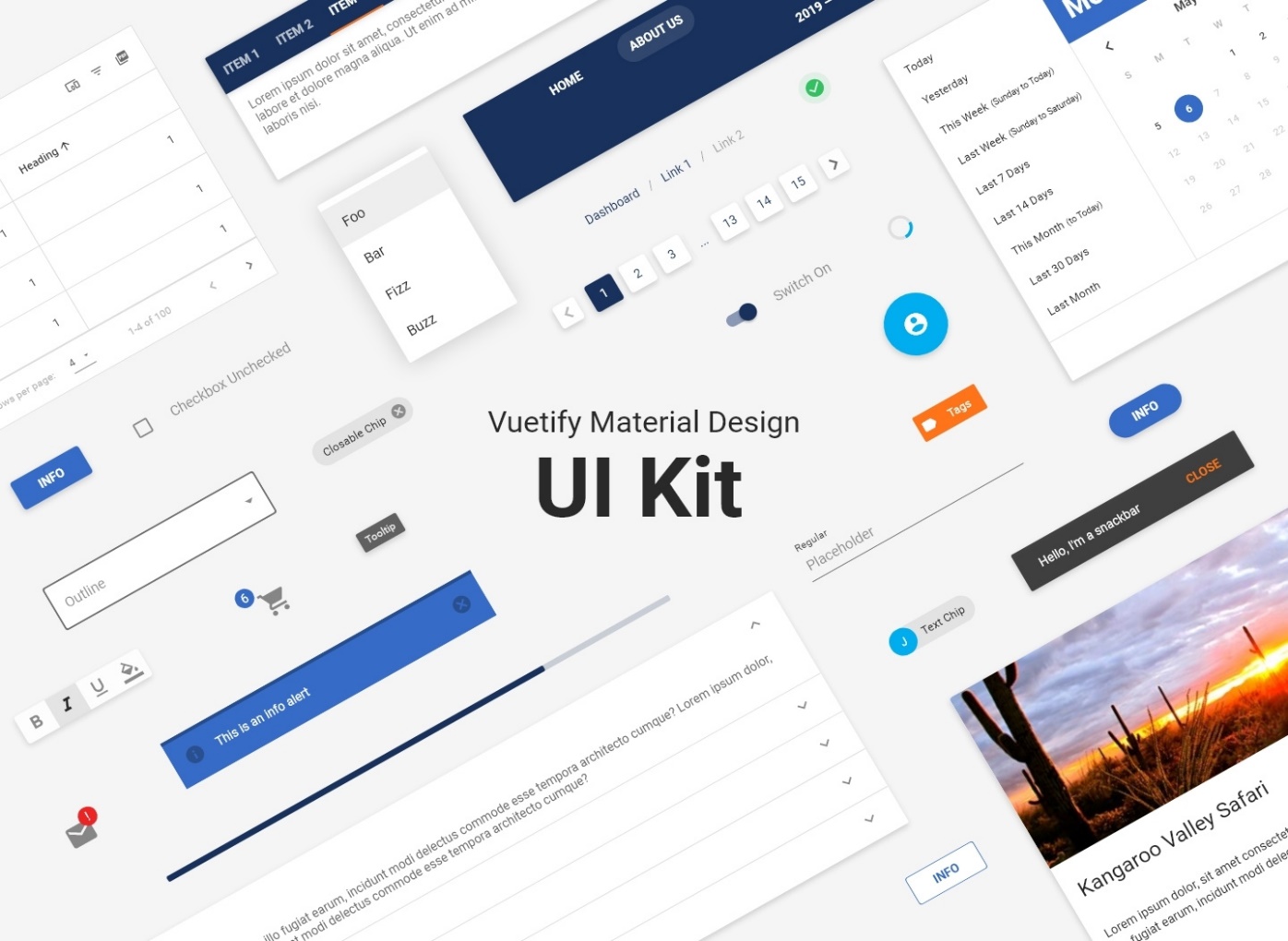
Kao i svaki framework Vue ima svoju predefinisanu biblioteku komponenti, koje se mogu pozivati unutar koda. Svaka komponenta ima svoj javascript, html i css, i moze se koristiti na više mjesta istovremeno.

Više informacija o ovome možete pronaći na VueJS oficijalnom websajtu <vuejs.org>, gdje je čitav framework detaljno opisan kroz njihovu dokumentaciju.

3. Vuetify

Vuetify predstavlja najbolju biblioteku za VueJS koja se aktivno razvija jos od 2016 godine.

Glavni cilj ovog projekta je da obezbijedi korisnicima sve potrebne komponente, dodatke, alate I plugine za olakšanu izradu modernih, responsivnih i brzih web aplikacija.

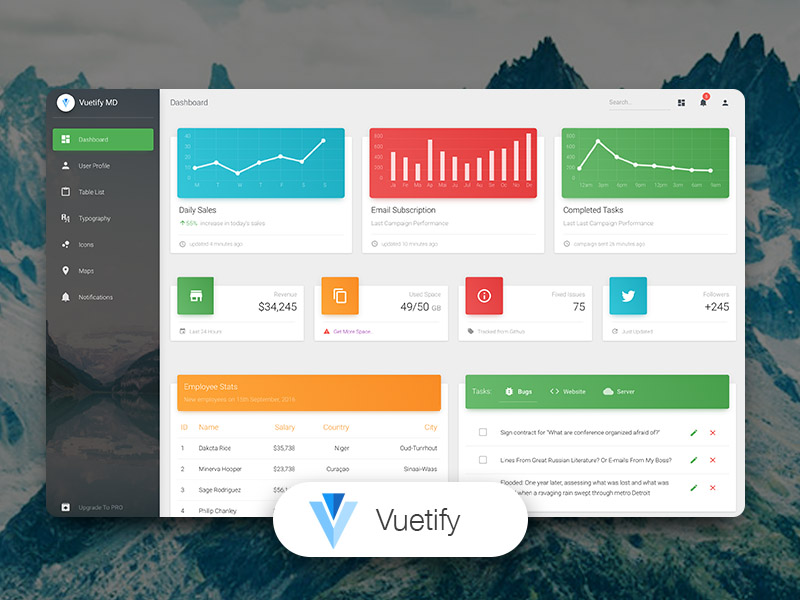


Slika 3.1 Vuetify Material Design UI Kit

U suštini Vuetify predstavlja obogaćenu kolekciju svih alata i komponenti koje koristi VueJS sa dodatnim funkcionalnostima.

Zbog svoje popularnosti Vuetify odlikuje bogat ekosistem alata i dodataka, nedeljna izdanja (update-i), masovnu zajednicu, profesionalne usluge i dugoročnu podršku.

Pored pojačanih Vue komponenti, neke od glavnih prednosti Vuetify biblioteke je ogromna zajednica od koje možete besplatno preuzeti mnoštvo profesionalno odrađenih tema za razne vrste web aplikacija.



Slika 3.2 Vuetify Material Dashboard template

Tema korištena u ovom projektu je upravo preuzeta sa jednog takvog mjesta. Zove se Vuetify Material Dashboard i besplatna je.

Napravljena je od strane Creative Tim-a i možete je preuzeti na sledećem linku: <https://www.creative-tim.com/product/vuetify-material-dashboard>

4. Problemi i implementacija OPTIMIZACIJE REUSABLE KODA U VUE.JS FRAMEWORKU

5. Zaključak

6. LITERATURA