



Fakultet informatike u Puli

Vue.js - Autentifikacija

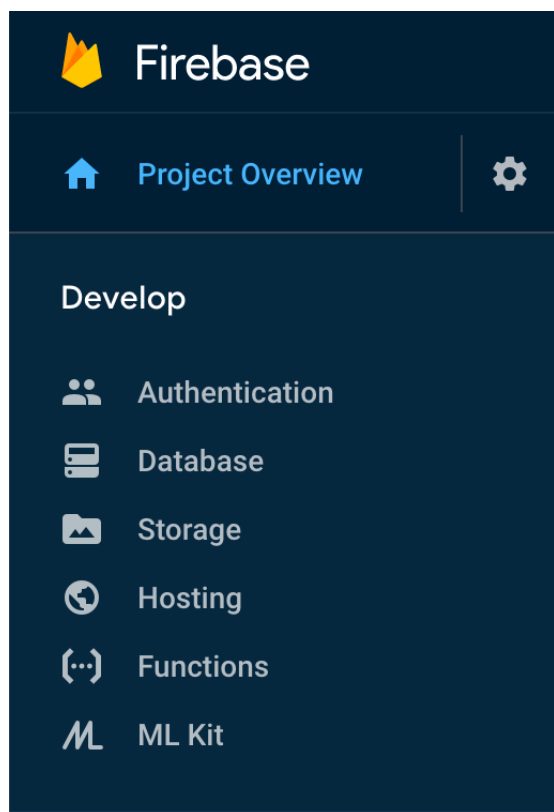
U ovoj vježbi prikazivat ćemo jedan od načina na koji se može ostvariti autentifikacija u aplikaciji. Pritom ćemo koristiti Firebase bazu i JS pakete.

Koraci

1. Kod sa prethodnih VUE-03 vježbi možemo pruezi s GitHuba. Repozitorij: <https://github.com/dbulic/instaclone> (branch `plan-wk3`). Preuzimanje s Git-a, instaliranje paketa i pokretanje aplikacije pojašnjeno je u prethodnim vježbama.
2. Za korištenje Firebase-a u JavaScript/Vue aplikacije potrebno je uključiti Firebase skripte. Skripte za zaglavlje možemo pronaći u Firebase dokumentaciji (From the CDN dio): <https://firebase.google.com/docs/web/setup#add-sdks-initialize>. Dodati ćemo ih na dno `body` elementa u `index.html` datoteku ispod `<div id="app"></div>`:

```
1    ...
2    <div id="app"></div>
3
4    <!-- Firebase App (the core Firebase SDK) is always required and must
be listed first -->
5    <script src="https://www.gstatic.com/firebasejs/7.5.2/firebase-
app.js"></script>
6
7    <!-- If you enabled Analytics in your project, add the Firebase SDK
for Analytics -->
8    <script src="https://www.gstatic.com/firebasejs/7.5.2/firebase-
analytics.js"></script>
9
10   <!-- Add Firebase products that you want to use -->
11   <script src="https://www.gstatic.com/firebasejs/7.5.2/firebase-
auth.js"></script>
12   <script src="https://www.gstatic.com/firebasejs/7.5.2/firebase-
firestore.js"></script>
13
14   </body>
15 </html>
```

- Time smo uključili potrebne **firebase** JavaScripte objekte u svoju aplikaciju. U ovom koraku otvorit ćemo novi Firebase račun (ukoliko ga već nemamo). Potrebno je izvršiti registraciju na <https://firebase.google.com> i ulogirati se na administracijsku konzolu: <https://console.firebase.google.com/u/0/> i stvoriti novi Firebase projekt (nije nužno uključiti Firebase analytics).
- Otvori svoj Firebase projekt i sa izbornika odaberi **Authentication**. Zatim odaberi **Set up sign-in method** i uključi (**Enable**) Email/Password način logina (za svoj projekt možeš podesiti i različite oblike logina poput Google ili Facebook logina).



Slika 1. Izbornik Firebase konzole.

- Još je potrebno u Vue projektu postaviti da se spaja na naš Firebase projekt. Do sada smo samo uključili Firebase biblioteke i omogućili autentifikaciju preko emaila i šifre. Podatke za spajanje na Firebase projekt možemo naći u Firebase konzoli u postavkama projekta (klik na zupčanik na izborniku) u sekciji **Firebase SDK snippet** i podsekciji **CDN** (ili **Config**).

Firestore SDK snippet

☐ Automatic  ☒ CDN  ☐ Config 

Copy and paste these scripts into the bottom of your <body> tag, but before you use any Firebase services:

```
<!-- The core Firebase JS SDK is always required and must be listed
<script src="https://www.gstatic.com/firebasejs/7.5.2/firebase-app.js"

<!-- TODO: Add SDKs for Firebase products that you want to use
      https://firebase.google.com/docs/web/setup#available-libraries

<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIzaSyDDxH7r2SbULMH4PftZQ4EJSUq0rEh9ZYQ",
    authDomain: "instaclone-ntankovic.firebaseio.com",
    databaseURL: "https://instaclone-ntankovic.firebaseio.com",
    projectId: "instaclone-ntankovic",
    storageBucket: "instaclone-ntankovic.appspot.com",
    messagingSenderId: "98968488013",
    appId: "1:98968488013:web:d2235bb46d4c788d7a3404"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```



Slika 2. Primjer Firebase konfiguracije.

Navedenu konfiguraciju kopiramo i smjestim na dno `<body>` elementa u `index.html` datoteku unutar novog `<script>` HTML elementa.

- Time je naša aplikacija povezana sa Firebaseom i možemo napraviti mogućnost registracije pomoću komponente `Signup.vue`. Prvo ćemo definirati Vue varijable za Email i dvije lozinke u `Signup.vue`:

```
1  <script>
2  export default {
3    data() {
4      return {
5        email: "",
6        password: "",
7        passwordAgain: ""
8      }
9    }
10 }
11 </script>
```

Zatim ćemo povezati navedene varijable sa `v-model` na odgovarajuća `input` polja u `<template>` sekciji:

```
1   <form>
2     <div class="form-group">
3       <label for="emailField">Email address</label>
4       <input v-model="email" type="email" class="form-control"
5 id="emailField" aria-describedby="emailHelp" placeholder="Enter email">
6       <small id="emailHelp" class="form-text text-muted">We'll never
7 share ...</small>
8     </div>
9     <div class="form-group">
10      <label for="passwordField">Password</label>
11      <input v-model="password" type="password" class="form-control"
12 id="passwordField" placeholder="Password">
13    </div>
14    <div class="form-group">
15      <label for="confirmPasswordField">Confirm Password</label>
16      <input v-model="passwordAgain" type="password" class="form-
17 control" id="confirmPasswordField" placeholder="Confirm password">
18    </div>
19    <button type="submit" class="btn btn-primary mt-5">Sign
20 up</button>
21  </form>
```

Potrebna nam je i nova metoda koja će učitati utipkane podatke i pozvati Firebase metodu za registraciju. Dodat ćemo je u `methods` sekciju:

```
1   export default {
2     name: "signup",
3     data() {
4       return {
5         email: "",
6         password: "",
7         passwordAgain: ""
8       }
9     },
10    methods: {
11      signup() {
12        firebase.auth().createUserWithEmailAndPassword(this.email,
13 this.password).catch(function(error) {
14          console.log(error);
15        });
16      }
17    }
18  }
```

Namjestit ćemo da se navedena metoda poziva prilikom *submits forme* (`<template>` dio u `Signup.vue`):

```

1    ...
2    <form @submit.prevent="signup">
3    ...

```

Ukoliko je sve u redu trebali bi se moći registrirati i vidjeti novoga korisnika u Firebase konzoli (sekcija Authentication):

Identifier	Providers	Created	Signed In	User UID ↑
nikola@tankovic.me		Dec 7, 2019	Dec 7, 2019	568GNwMShZc71A8Ev39cGeDr2g...

Rows per page: 50 1-1 of 1

- Na sličan način podesiti `Login.vue` komponentu. Potrebno je umjesto Firebase metode za *sign-up* koristiti metodu za *sign-in*. Možeš ju pronaći u Firebase dokumentaciji :) P.S. Razmisli jesu li potrebni baš identične varijable u `data` sekciji.
- Registracija i Login sada rade. Međutim aplikacija je isto nezavisno o tome jeli trenutni korisnik ulogiran ili ne. U ovom ćemo koraku:

- razlikovati kada je korisnik ulogiran i prema tome prikazati ili sakriti određene dijelove (npr. Login button),
- nakon *logina* ili *logouta* preusmjeriti korisnika na `/home` ili `/login`.

Da bi pratili stanje autentifikacije korisnika moramo dodati tzv. *listener* koji reagira na promjenu stanja autentifikacije (<https://firebase.google.com/docs/reference/js/firebase.auth.Auth.html#onauthstatechanged>).

U glavnu komponentu `App.vue` dodajemo sljedeći isječak:

```

1    firebase.auth().onAuthStateChanged(user => {
2      if (user) {
3        // User is signed in.
4        console.log(user.email);
5      }
6      else {
7        // User is not signed in.
8        console.log("No user");
9      }
10   });

```

U `store.js` već imamo varijablu `authenticated` pomoću koje prikazujemo/sakrivamo *button* Login/Logout. Podesi `store.authenticated` na `true/false` zavisno o stanju autentifikacije u gornjem isječku. Također, u slučaju autentificiranog korisnika podesi `store.userEmail` na e-mail korisnika.

⚠ Razmisli: zašto smo praćenje autentifikacije dodali u `App.vue`, a ne recimo u `Login.vue` ?

Još je potrebno napraviti *logout* akciju. Već imamo `logout` metodu u koju treba pozvati `Firebase` metodu za logout. Pronađi ju u dokumentaciji i zamijeni kod u postojećoj metodi. Zašto nije potrebno ostaviti `store.authenticated = false` ?

9. U zadnjem koraku dodat ćemo još redirekciju nakon logina i logouta. Razmisli gdje ćeš dodati sljedeće naredbe za redirekciju na `/login` stranicu i redirekciju na `/` (Home) stranicu:

```
1 // redirekcija na "home" stranicu (naziv stranice "home" definiran je u
  routes/index.js)
2 if (this.$route.name !== 'home')
3   this.$router.push({name: 'home'})
4
5 // redirekcija na "login" stranicu
6 if (this.$route.name !== 'login')
7   this.$router.push({name: 'login'})
```

10. **Samostalni zadatak.** Provjeriti jesu li dvije lozinke iste pri registraciji. Napraviti novu stranicu postavki korisnika. Na toj stranici omogućiti korisniku promjenu lozinke. Zadatak predati na predviđeno mjesto pod šifrom VUE-04.