



# Vue.js - Spremanje dodatnih korisničkih podataka

U ovoj vježbi prikazivat ćemo mogućnost spremanja dodatnih korisničkih podataka uz registraciju.

## Koraci

1. Kod sa prethodnih VUE-05 vježbi možemo preuzeti s GitHuba. Repozitorij: <https://github.com/fipu-nastava/fipugram> (branch `step6`). Preuzimanje s Git-a, instaliranje paketa i pokretanje aplikacije pojašnjeno je u prethodnim vježbama.
2. Prvo ćemo dodati prikazivanje greške prilikom registracije u `Signup.vue`. Koristit ćemo Bootstrap traku za prikaz grešaka:

```
1 <div class="alert alert-danger">
2   <strong>Ups!</strong>
3   Došlo je do greške.
4 </div>
```

Kako bi spriječili da se navedena greška stalno prikazuje, postaviti ćemo novu varijablu u `data()` i pomoću `v-if` direktive kontrolirati prikaz i sadržaj greške:

```
1 ...
2 <div v-if="errorMessage" class="alert alert-danger">
3   <strong>Ups!</strong>
4   {{ errorMessage }}
5 </div>
6 ...
7
8 <script>
9 import store from "../store";
10 export default {
11   data() {
12     return {
13       errorMessage: "", // defaultno prazno, pa se greška ne prikazuje
14     }
15   }
16 }
```

3. Navedenu grešku postaviti ćemo u `.catch()` dijelu Firebase metode za registraciju:

```

1    //...
2    methods: {
3      signup() {
4        firebase
5          .auth()
6          .createUserWithEmailAndPassword(this.email, this.password)
7          .then(() => {
8            // ...
9          })
10         .catch(error => {
11           console.error(error);
12           this.errorMessage = error.message; // postavljanje poruke s
greškom
13         });
14       }
15     }
16   };

```

Na taj način prikazat će se pogreška prilikom registracije.

4. Sada ćemo dodati mogućnost da korisnik odabere vrstu profila. Pritom će nam poslužiti `<select>` element koji opcije povlači iz varijable `tipProfila`. Prvo ćemo dodati `tipProfila: ["Javni", "Privatni"]`, u `data()` a zatim novi element u formu koji odabrani profil sprema u varijablu `odabraniTipProfila` (također ju treba dodati u `data()`):

```

1    <div class="form-group">
2      <label for="tipProfila">Tip profila</label>
3      <select v-model="odabraniTipProfila"
4        id="tipProfila"
5        class="form-control form-control-lg">
6        <option v-for="k in tipProfila">{{k}}</option>
7      </select>
8    </div>

```

Zatim ćemo modificirati metodu za registraciju korisnika koja će nakon uspješne registraciju u novu kolekciju `users` spremiti dodatne podatke o korisniku. Kao `id` zapisa možemo koristiti korisnički `email` jer je jedinstven:

```

1    methods: {
2      signup() {
3        firebase
4          .auth()
5          .createUserWithEmailAndPassword(this.email, this.password)
6          .then(() => {
7            // postavi podatke o korisniku
8            let id = this.email;
9            // sada moramo spremiti te dodatne podatke
10           db.collection("users")
11             .doc(id)
12             .set({
13               tipProfila: this.odabraniTipProfila

```

```

14         })
15         .then(function() {
16             console.log("Document successfully written!");
17         })
18         .catch(function(error) {
19             console.error("Error writing document: ", error);
20         });
21     })
22     .catch(error => {
23         console.error(error);
24         this.errorMessage = error.message;
25     });
26 }
27 }

```

5. Ono što nam je još potrebno jest dohvatiti te dodatne podatke o korisniku nakon uspješne prijave. To možemo načiniti u komponenti `App.vue` nakon provjere da je korisnik autentificiran:

```

1  //...
2  if (user) {
3      self.userEmail = user.email;
4      self.authenticated = true;
5
6      // novi kod:
7      db.collection("korisnici")
8      .doc(user.email)
9      .get()
10     .then(doc => {
11         if (doc.exists) {
12             console.log("Document data:", doc.data());
13             this.tipKorisnika = doc.data().tipProfila;
14         } else {
15             // doc.data() will be undefined in this case
16             console.log("No such document!");
17         }
18     });
19     //...

```

Dodatni podatak sprema se u `tipKorisnika` koji je potrebno inicijalizirati u `store.js`.