

## 2. naloga - Oblikovanje testnih primerov

### 1. Funkcionalnost: prijava v sistem

Test:	LoginForm (LoginForm.test.js)
Predpogoji:	Nahajamo se na login strani
Vhodni podatki:	Validno uporabniško ime in geslo
Izhodni podatki:	Uspešna prijava in reusmeritev na seznam vseh uporabnikov
Test uspešen?:	DA
Koraki v testu:	<ol style="list-style-type: none"><li>1. Login stran se uspešno naloži</li><li>2. Če so polja prazna se pojavi alert</li><li>3. Če vpišemo nepravilne podatke se pojavi alert</li><li>4. Ob uspešni prijavi smo preusmerjeni na url: /employees</li></ol>

### 2. Funkcionalnost: Registracija v sistem

Test:	RegisterForm (RegisterForm.test.js)
Predpogoji:	Nahajamo se na register strani
Vhodni podatki:	Vnesti je potrebno uporabniško ime, priimek, elektronski naslov in geslo
Izhodni podatki:	Preusmeritev na url: /login
Test uspešen?:	DA
Koraki v testu:	<ol style="list-style-type: none"><li>1. Register stran se uspešno naloži</li><li>2. Če so polja prazna se pojavi alert</li><li>3. Če vpišemo nepravilne podatke se pojavi alert</li><li>4. Ob uspešni prijavi smo preusmerjeni na url: /login</li></ol>

### 3. Funkcionalnost: Filtriranje zaposlenih

Test:	EmployeeListSearch (EmployeeListSearch.test.js)
Predpogoji:	<ol style="list-style-type: none"><li>1. Smo uspešno prijavljeni v sistem. Ni potrebno biti admin, lahko smo katerikoli profil.</li><li>2. Test more imeti napisane podljubne mockup zaposlene</li></ol>
Vhodni podatki:	V vhod vnesemo ime ali/in priimek od zaposlenega
Izhodni podatki:	Tabela z vsemi zaposlenimi se bo prefiltrirala ter prikazala samo tiste zaposlene, ki jih iščemo
Test uspešen?:	DA

Koraki v testu:	<ol style="list-style-type: none"> <li>1. Ob vnosu 'NonExistingUser' se prikaže prazna tabela, saj nič ne ustreza vnosu</li> <li>2. Vnesemo ime in/ali priimek ter ta uporabnik po prikazan v tabeli</li> </ol>
-----------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### 4. Funkcionalnost: Urejanje zaposlenih

Test:	EmployeeList (EmployeeList.test.js)
Predpogoji:	<ol style="list-style-type: none"> <li>1. Prijavljeni moramo biti kot admin!</li> <li>2. Test more imeti napisane podljubne mockup zaposlene</li> </ol>
Vhodni podatki:	/
Izhodni podatki:	Arhiviranje, brisanje ali spreminjanje Zaposlenega
Test uspešen?:	DA
Koraki v testu:	<ol style="list-style-type: none"> <li>1. Tabela se pravilno naloži</li> <li>2. Preverimo, če je število zaposlenih enako številu delete gumbov</li> <li>3. Izbrišemo zaposlenega</li> <li>4. Arhiviramo zaposlenega</li> <li>5. Spremenimo podatke o zaposlenem</li> </ol>

#### 5. Funkcionalnost: Prikaz profila zaposlenega

Test:	EmployeeProfile (EmployeeProfile.test.js)
Predpogoji:	3. Prijavljeni moramo biti kot katrrikoli uporabnik (mockup za teste)
Vhodni podatki:	Podatki o prijavljenem uporabniku
Izhodni podatki:	Prikaz podatkov o uporabniku na njegovi strani
Test uspešen?:	DA
Koraki v testu:	<ol style="list-style-type: none"> <li>1. Od uporabnika stran se uspešno naloži</li> <li>2. Uporabnik pritisne na gumb “View my profile”</li> <li>3. Preusmerjeni smo na url: /profile</li> <li>4. Na novi strani so prikazani vsi podatki o uporabniku</li> </ol>