

Job interview test exercixe

Név: Szemesi Gábor

Email: gszemesi@gmail.com

Dátum: 2023.05.28

Feladat

Webes környezetben űrlap készítése. Az oldalon egy vállalat és annak dolgozói adatait lehet felvenni. Az oldalon a vállalat adatainál megadott munkavállaló számmal megegyező munkavállaló terület jelenik meg. Az űrlapon megadott adatoknak érvényességét vizsgálni kell. Ha minden adat helyes, akkor az oldal alján megjelennek az űrlapban megadott adatok.

Űrlap érvényesítése:

- vállalat neve és e-mail címe kötelező mezők, amelyeket ki kell tölteni
- vállalat e-mail-címnek érvényes e-mail-címnek kell lennie
- alkalmazottak számának pozitív egész számnak kell lennie, legalább 1 és legfeljebb 100
- az alkalmazott neve, e-mail címe, beosztása és életkora kötelezően kitöltendő mezők
- az alkalmazotti e-mail-címnek érvényes e-mail-címnek kell lennie
- a munkavállaló életkorának pozitív egész számnak kell lennie, és legalább 18 évesnek kell lennie
- az alkalmazotti önéletrajznak pdf fájlnek kell lennie

Feladat elemzése

A feladat két részre osztható. Egyik a vállalat adatai másik az alkalmazottak adatai. A vállalat adatainak helyes kitöltése után a „Render” gombbal létre hozható a megfelelő számú alkalmazotti űrlap. Ha valamelyik input érvénytelen akkor piros hiba üzenet jelenik meg az input mező alatt. Ha az alkalmazottak input adatainál is van érvénytelen, akkor ott is piros hiba üzenet jelenik meg a „Save” gomb megnyomása után. Ha viszont minden adat helyes, akkor az oldal alján megjelennek az űrlapból begyűjtött adatok.

Osztályok, metódusok és megvalósítás

Az űrlap egy React alkalmazás segítségével valósul meg. A React alap objektumain kívül két fájlt hozok létre: Company.js és Employee.js. Az objektumok saját maguk felelősek az adatok érvényességének vizsgálatának, majd az adatok elküldése az App.js-nek.

App.js	callbackCompany	A Company.js által küldött vállalati adatokat menti el
	callbackEmployee	A Employee.js által küldött alkalmazottak adatait menti el
	save	Megváltoztatja setPrint értékét és így láthatóak lesznek az adatok az oldal alján.
	[isPrint, setPrint]	boolean, kiírás megtörténhet-e
	[refresh_value, refresh]	boolean, az oldalon lévő adatok frissítéséhez felvett érték (értéke független mindentől)
	[showError, setShowError]	boolean, a „Save” gomb lenyomása után megjelennek az esetleges hiba üzenetek az alkalmazotti objektumokban

Company.js	handleSubmit	"Render" gomb megnyomására megvizsgálja az értékeket és ha helyes elküldi az App.js-nek
Employee.js	objektum belsejében lefutó ha feltétel	ha nincs hiba, akkor megpróbálja elküldeni az App.js-nek az adatokat

A Company.js és Employee.js-ben felvett változók és függvények triviálisak.

Tesztelési terv és esemény kezelés

A „Render” gomb tesztelése/kiváltott esemény: gombnyomásra megjelenik pontosan annyi alkalmazotti objektum, mint ami az alkalmazotti szám mezőben meglett adva.

A „Save” gomb tesztelése/kiváltott esemény: gombnyomásra megjelenik az oldal alján az űrlapban megadott adatok.

Az input adatokat tesztelése is szükséges. Csak a helyes adatokat fogadja el a rendszer és a hiba üzenet megjelenik és megfelelő hiba üzenet jelenik-e meg.

pl.: Company inputjainak teszt esete

Mit tesztelünk:	Teszt eset (input):	Elvárt kiment:
vállalat neve	""	„Name can't be Empty” hiba üzenet megjelenése
vállalat neve	"Budapest Kft. "	Nem jelenik meg semmilyen hiba üzenet
vállalat email címe	""	„Email can't be Empty” hiba üzenet megjelenése
vállalat email címe	"ez nem egy email cím"	„Email has to valid” hiba üzenet megjelenése
vállalat email címe	"minta@minta.hu"	Nem jelenik meg semmilyen hiba üzenet
alkalmazottak száma	0	„Number of employee can't be this numb” hiba üzenet megjelenése
alkalmazottak száma	1	Nem jelenik meg semmilyen hiba üzenet
alkalmazottak száma	100	Nem jelenik meg semmilyen hiba üzenet
alkalmazottak száma	101	„Email can't be Empty” hiba üzenet megjelenése

További módosított fájlok

index.html (header hozzá adása), index.css, App.css