

Završni ispit iz Programiranja za veb –Jun 2 2020/21 – 1. grupa

03.07.2021.

Ispit se polaže najviše 180 minuta. Raspodela poena je data u narednoj tabeli:

	Zadaci		Teorija										
Redni broj	1	2	1	2	3	4	5	6	7	8	9	10	Suma
Najviše poena	15	20	5	5	5	5	5	5	5	5	5	5	70*

*Poeni za teoriju se skaliraju sa ukupnih 50 na 35.

Da bi se položio završni ispit, potrebno je ostvariti najmanje: 5 poena na Node.js zadatku, 10 poena na Angular zadatku i 12 poena na teorijskom delu. Rešenja zadataka koja imaju sintaksnih grešaka nose 0 poena. Kod pod komentarima neće biti pregledan.

Teorijska pitanja

Uputstvo: U radnom direktorijumu na *Desktop*-u nalazi se datoteka *PrezimeIme_alasNalog_grupa_teorija.txt* u kojoj se nalaze teorijska pitanja. U ovoj datoteci ćete smestiti odgovore na pitanja. Obavezno preimenovati datoteku da sadrži Vaše informacije. Obavezno pratiti sve instrukcije koje su date u datoteci.

Napomena za zadatke

Ukoliko neki zahtevi u tekstovima zadataka mogu da se urade na više načina, onda:

- Ako je tekst u zahtevu **podebljan**, to znači da je potrebno da se taj zahtev implementira na opisan način.
- U suprotnom, ostavljeno je Vama da odlučite na koji način ćete implementirati. Slično, ako neki deo zahteva nije eksplicitno naveden (na primer, ako nije naglašeno da se neki formular validira ili da se vrši obrada u slučaju grešaka), onda se podrazumeva da ne morate da vodite računa o tome u Vašim rešenjima.

Zadaci

Implementirati aplikaciju koja omogućava pregledanje rečnika engleskog jezika.

1. Napisati Node.js aplikaciju koja ispunjava naredne zahteve:

- U MongoDB bazi podataka *pueb_jun2_2021* nalazi se kolekcija *words* koja sadrži informacije o pojedinačnim rečima. Sledeća polja se koriste za opis jedne reči:
 - jedinstveni identifikator (*_id*),
 - reč (*word*),
 - značenje reči (*meaning*),
 - nivo znanja jezika kojem reč pripada (*level*)
 - niz sa primerima upotrebe reči (*examples*).

Korišćenjem Mongoose biblioteke kreirati shemu koja prati opisanu kolekciju. Sva polja u shemi postaviti na **obavezna** i **ograničiti** dužinu same reči *word* i njenog značenja *meaning*, redom, na 50 i 100 karaktera. Potom kreirati model *Word*.

- Korišćenjem Express i Mongoose biblioteka implementirati naredni REST API:
 - GET /api/words
 - Značenje: dohvaćanje svih reči iz rečnika.
 - Odgovor: JSON reprezentacija rezultujućeg niza i statusni kod 200.
 - GET /api/words/{level}
 - Značenje: dohvaćanje svih reči koje pripadaju naznačenom nivou. Dozvoljene vrednosti za nivo su slova *A*, *B* i *C*.
 - Odgovor: JSON reprezentacija rezultujućeg niza ili *null* ukoliko naznačeni nivo ne postoji ili nije naveden. Statusni kod prilagoditi uspešnosti zahteva.

– PUT /api/words

- * Značenje: ažuriranje primera upotrebe reči. U telu zahteva se prosledjuje identifikator reči i niska koja predstavlja novi primer upotrebe reči.
- * Odgovor: JSON reprezentacija ažuriranog objekta ili *null* ukoliko nedostaje neki od podataka. Statusni kod prilagoditi uspešnosti zahteva.

- Voditi računa da se odgovarajući elementi serverske aplikacije nalaze u odgovarajućim datotekama.
- Sve greške proslediti funkciji srednjeg sloja koja radi na nivou grešaka (definisanoj u *app.js* datoteci).

2. Napisati jednostraničnu Angular aplikaciju koja ispunjava naredne zahteve**:

- Slanje svih HTTP zahteva koji su navedeni ispod implementirati u jednom **Angular servisu**. Obezbediti postojanje tačno jedne instance ovog servisa na nivou cele aplikacije.
- Kreirati komponentu *WordsComponent* koja na adresama *http://localhost:4200/* ili *http://localhost:4200/words* u formi tabele prikazuje reči iz rečnika (slika 1). Reči iz rečnika se dohvataju slanjem odgovarajućeg zahteva servisa. Klikom na reč prikazuje se sekcija koja sadrži dodatne informacije o reči: njen nivo i sve primere upotrebe. Ovaj ispis prati i **reaktivni** formular koji omogućava unos novog primera za odabranu reč (slika 2).

Primer upotrebe reči je korektan ukoliko počinje velikim slovom i završava se nekim interpunkcijskim znakom (tačkom, upitnikom ili uzvičnikom). Napisati validacionu funkciju koja vrši ovu proveru i pridružiti je polju za unos primera. Dodati i validaciju koja ovo polje čini obaveznim, kao i validaciju kojom se proverava da li je dužina kraća od 100 karaktera. Prilikom dodavanja novog primera, u slučaju pogrešnog unosa prikazati porukicu o grešci (slika 3). Ukoliko je primer korektan, dodavanje realizovati slanjem odgovarajućeg zahteva servisa. Potom ispisati porukicu o uspešnosti (slika 4), ažurirati primere upotrebe reči (slika 5) i resetovati formular.

Na dnu tabele koja sadrži sve reči rečnika nalazi se serija linkova koja omogućava pregledanje reči odgovarajućeg nivoa.

Za stilizovanje tabele mogu se koristiti Bootstrap klase *table* i *table-striped*. Za stilizovanje linkova pridruženih rečima može se koristiti klasa *link-primary*. Za stilizovanje formulara mogu se koristiti klase *form-group*, *form-control* i *btn-primary*. Ulogu labele pridruženu polju formulara ima *tekst-čuvar mesta*.

- Kreirati komponentu *WordsByLevelComponent* koja prikazuje informacije o svim rečima (slika 6) zadatog nivoa na adresama oblika *http://localhost:4200/words/{level}*. Reči se prikazuju u formi tabele koja je organizovana na isti način kao i tabela komponente *WordsComponent* uz naslov koji je prilagodjen nivou reči. Ukoliko se u putanji navede parametar koji ne odgovara ni jednom od nivoa A, B ili C prikazati odgovarajuću poruku (slika 7). Nivoa je moguće zadavati i malim slovima.
- Voditi računa o curenju memorije.

**Da bi Vam bile dostupne klase koje su navedene iznad, u datoteci *klijent/src/index.html* u sadržaju elementa *<head>* potrebno je da dodate naredne linije:

```
<link rel="stylesheet" type="text/css" href="/assets/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="/assets/styles.css">
```

Dictionary

word	meaning
explore	to search and discover (about something)
brownout	an occasion when less electric power than usual is supplied to an area
coherent	having its parts related in an organized and reasonable way
data	information collected for use
affect	to have an influence on someone or something
book	a written text that can be published in printed or electronic form
cloud	a gray or white mass in the sky made up of very small floating drops of water
honey	a sweet, sticky, yellow substance made by bees and used as food
music	a pattern of sounds made by musical instruments, voices, or computers

Show words by levels: [A](#) , [B](#) , [C](#)

Slika 1: Prikaz svih reči rečnika - komponenta WordsComponent.

Word level: A

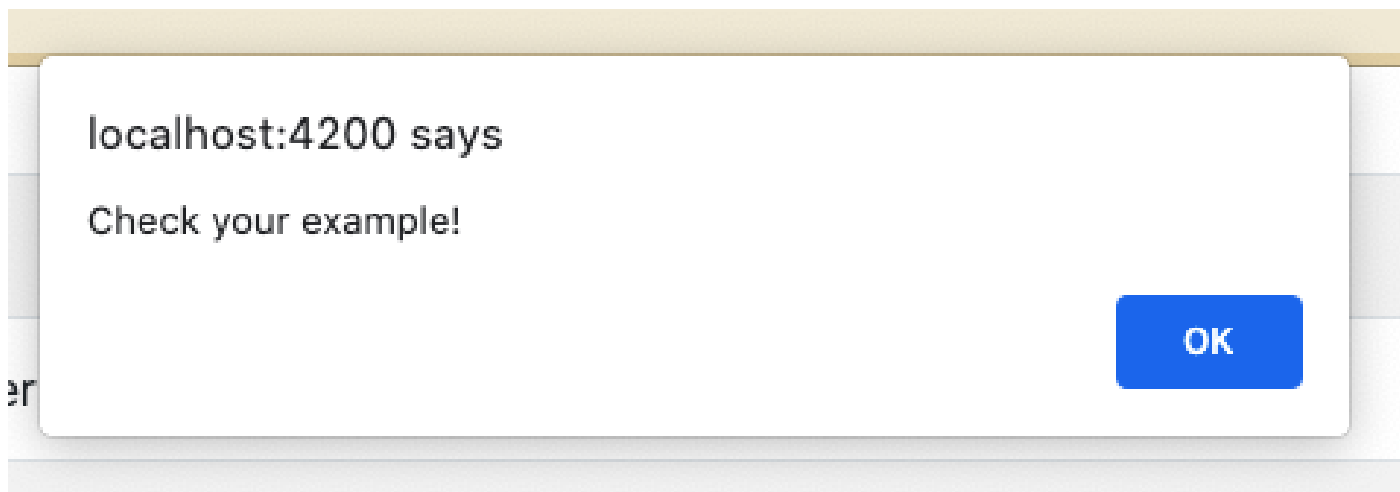
Examples:

- Have you read any good books recently?
- He has a new book out.

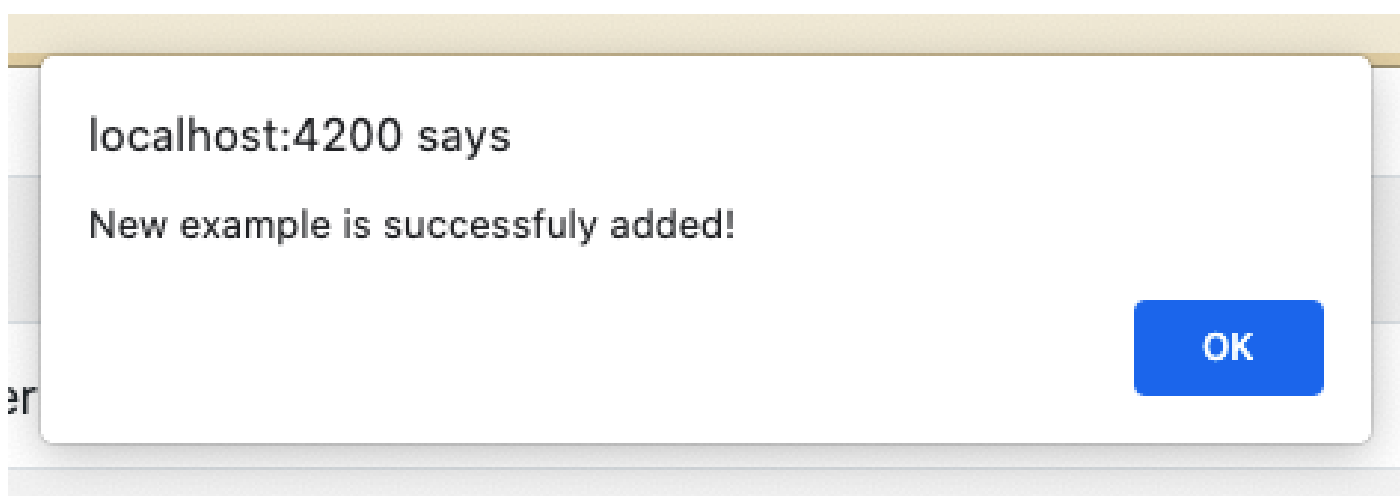
Share your example!

Add new example

Slika 2: Prikaz detaljnih informacija za odabranu reč *book* i prateći reaktivni formular.



Slika 3: Prikaz poruke o greši za primer u pogrešnom formatu.



Slika 4: Prikaz poruke nakon uspešnog dodavanja primera.

Word level: A

Examples:

- Have you read any good books recently?
- He has a new book out.
- This is a great book!

Share your example!

Add new example

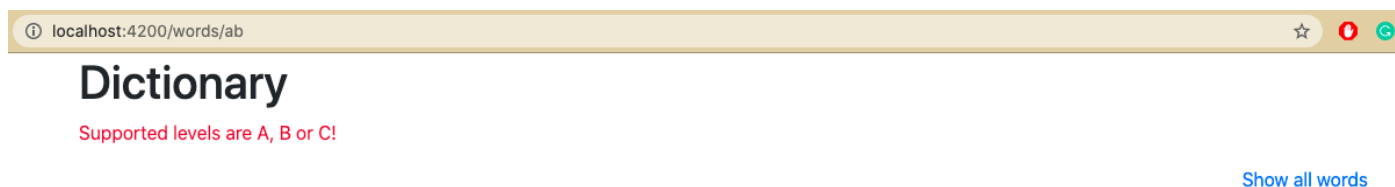
Slika 5: Prikaz ažurirane liste primera za reč *book*.

Dictionary - B level words

word	meaning
explore	to search and discover (about something)
data	information collected for use
affect	to have an influence on someone or something

Show all words

Slika 6: Prikaz reči odabranog nivoa - komponenta WordsByLevelComponent.



Slika 7: Prozorčić sa porukom o grešci u slučaju pogrešnog nivoa *ab*.