

Specifikacija prvog projekta iz predmeta Softverske komponente - Školska 2023/2024. godina

Implementacija komponente za rad sa rasporedom

Osmisliti i implementirati biblioteku (komponentu) koja će se koristiti za upravljanje podacima o rasporedu sa dve različite implementacije. Komponenta treba da ima odvojenu specifikaciju (API) u kome se nalaze sve operacije nad rasporedom kao što je opisano u nastavku. Specifikacija treba da ima dve implementacije koje su realizovane kao posebne biblioteke i razlikuju se u unutrašnjoj reprezentaciji podataka o rasporedu. Navedena specifikacija projekta nije u potpunosti dorečena i konkretna, na mnogim mestima je ostavljen prostor da se nađu kreativna rešenja i na proizvoljan način realizuju navedene ideje.

Opis funkcionalnosti komponente (API)

Raspored sadrži termine u dve dimenzije, vremenu i prostoru, a svaki termin može imati i vezane podatke, na primer za raspored časova na fakultetu ti podaci bi bili naziv predmeta, nastavnik koji drži čas, tip časa (predavanja ili vežbe), grupe koje slušaju čas. Komponenta treba da bude uopštena i proširiva tako da za termin mogu da se vežu proizvoljni podaci. Prostor u rasporedu ima podatke o kapacitetu i opremi koju poseduje, broj računara, projektor, grafička tabla, a ostaviti mogućnost da mogu da se vežu i još neki dodatni podaci. Termini mogu da imaju različito trajanje.

Kreiranje rasporeda

- inicijalizacija rasporeda
- dodavanje prostorija sa osobinama (kapacitet, računari, projektor)
- dodavanje novog termina uz provere o zauzetost, obraditi situaciju da je termin već zauzet
- brisanje zauzetog termina
- premeštanje termina - brisanje i dodavanje novog termina sa istim vezanim podacima

Pregled (pretraživanje) rasporeda

Najbitnije operacije nad rasporedom su provera zauzetosti termina i prostora i izlistavanje slobodnih termina po različitim kriterijumima. Vreme se prilikom ovih provera može zadavati na dva načina, prvi je zadavanje tačnog datuma, a drugi je zadavanje dana u nedelji i perioda (na primer da li je slobodan termin sredom 10-12h u periodu od 1.10.2023. do 1.12.2023). Termini se mogu zadavati kao vreme početka i završetka ili kao vreme početka i trajanje. Izlistavanje slobodnih termina može da uključi i tačnu prostoriju, prostoriju sa određenim osobinama (na primer učionica sa računarima, projektorom,

da ima više od 30 mesta i slično), a može i da bude nezavisno od prostorije (izlistati sve učionice koja je slobodna tog i tog dana). Obezbediti i pretraživanje rasporeda prema vezanim podacima (na primer ako je nastavnik vezani podatak, izlistati sve termine tog nastavnika ili sve termine kada je on slobodan).

Potrebno je obezbediti operacije za izlistavanje slobodnih termina, ali i zauzetih termina, provera da li je određeni termin slobodan ili zauzet, po raličitim kriterijumima kako je prethodno objašnjeno.

Učitavanje i snimanje rasporeda u fajl

- osmisлити i implementirati različite operacije za učitavanje podataka o rasporedu iz fajla različitih formata, na primer json, csv
- model rasporeda je uopšten, jedino što termin u rasporedu obavezno mora da sadrži je vreme i mesto, ostali podaci su proizvoljni, tako da će za ove operacije morati da se smisli i implementira neka vrsta konfiguracije (na primer, ako se čita iz csv-a da zaglavlje sadrži podatke iz rasporeda, kao što su na primer naziv predmeta, nastavnika, i sl i na kojim indeksima se nalaze podaci o terminu, dan u nedelji, oznaka učionice)
- u operacijama za čitanje iz fajla će biti potrebno da se zadaju i datumi početka i završetka važenja rasporeda, kao i izuzeti dani
- obezbediti snimanje rasporeda u fajlove csv, json i pdf formata sa različitim konfiguracijama, na primer snimanje rasporeda za određeni period, svi datumi posebno ili grupisani po danima u nedelji, samo za određeni podatak iz rasporeda (na primer za jedan predmet)

Prva implementacija - raspored kao kolekcija termina

Raspored se čuva kao kolekcija konkretnih termina u vremenu i prostoru. Primer jednog termina je sreda, 18.10.2023. 10-12h, soba S1.

Druga implementacija - raspored na nedeljnom nivou u datom periodu

Raspored se čuva na nedeljnom nivou za zadati period, na primer termin je soba S1, petak od 13-15h u periodu od 1.10.2023. do 20.1.2023. Ovakvo čuvanje rasporeda u stvari grupiše termine kao ponavljajuće događaje za određeni dan u nedelji. U zadatom periodu mogu postojati i izuzeti dani, kada ne važi dati raspored definisan na nedeljnom nivou. Ako je neki termin zauzet samo jedan dan, na primer četvrtak 19.10.2023. 12-15h, onda se on čuva kao četvrtak 12-15h u periodu 19.10.2023. do 19.10.2023.

Program koji koristi raspored

Pored navedenih biblioteka potrebno je implementirati program za komandnu liniju koji koristi specifikaciju i jednu implementaciju rasporeda. Program treba da podrži operacije za rad sa rasporedom časova na RAF-u. U programu se može učitati csv fajl sa rasporedom časova na RAF-u (skinuti fajl sa servisa), podaci o učionicama (napraviti txt fajl koji sadrži spisak svih učionica, kapacitet i da li ima računare). Prilikom učitavanja rasporeda zadaje se period važenja rasporeda, kao i izuzeti dani, osmisлити kako ovo da se zadaje, da li unosom preko konzole ili učitavanjem iz fajla. Obezbediti komande za pregled

i pretraživanje rasporeda po različitim kriterijumima, zauzimanje novog termina, prebacivanje termina, kao i snimanje rasporeda u fajl, za grupu, učionicu, nastavnika, ceo raspored i sl.

Implementacija programa za komandnu liniju treba da se oslanja samo na specifikacionu komponentu, a da se odgovarajuća implementaciona komponente veže u toku izvršavanja (runtime dependency).

Raspodela poena

1. Dobro osmišljena specifikacija, kvalitet implementacije specifikacije - 8p

2. Implementacija dve komponente koje implementiraju specifikaciju

 raspored kao kolekcija termina - 5

 raspored na nedeljnom nivou - 5

(raspodela poena po zahtevima - kreiranje rasporeda - 20%, pregled (pretraživanje) rasporeda 40%, učitavanje i snimanje rasporeda - 40%, procenti se odnose na specifikaciju i dve implementacije, na primer ako ne uradite učitavanje i snimanje, a sve ostalo kompletno, možete dobiti maks 11 poena)

3. Kvalitetna dokumentacija za specifikacionu komponentu - 1p

4. Implementacija programa za komandnu liniju - 3p

5. Automatizacija pakovanja biblioteka i programa za komandnu liniju, generisanje dokumentacije korišćenjem build alata - 3p

Napomene

Preporučeni jezik za implementaciju je Java, a alati za izgradnju Apache Maven ili Gradle.

Ukoliko tim odluči da ne radi program za komandnu liniju, potrebno je napraviti neki testni program bez komunikacije sa korisnikom, program treba da ilustruje rad implementiranih komponenti, ovaj program ne nosi poene, ali se mora napraviti da bi se projekat ocenio.

Bild alatom treba automatizovati pakovanje aplikacije za komandnu liniju tako da ona može bez dodatnih podešavanja da se pokreće izvan razvojnog okruženja (Eclipse, IntelliJ), proces pakovanja i pokretanja aplikacije se demonstrira na odbrani. Na odbrani student mora da pokaže razumevanje bild procesa i konfiguracije dipendensija u bild alatu. Opisani zahtevi su uslov da bi student ostvario poene na 5. tački.

Rok za predaju i odbrana:

Projekat se radi u timu od dvoje i rok za završetak je termin odbrane, projekat se donosi na odbranu i pokazuje ispitivaču, nije potrebno slati ranije.

Projekat se brani u kolokvijumskoj nedelji, nedelju dana pre termina odbrane biće organizovana prijava timova za odbranu, a raspored odbrane će biti objavljen na materijalima. Studenti mogu, ako žele da projekat brane i ranije u dogovoru sa asistentima.

Članovi tima se posebno ocenjuju prema procenjenom uloženom trudu.