

Specifikacija prvog projekta iz predmeta Softverske komponente - školska
2019/2020. godina

Implementacija komponente za skladištenje fajlova

Opis projekta:

Osmisliti i implementirati biblioteku (komponentu) koja će se koristiti za skladištenje različitih vrsta fajlova sa opcionim unosom metapodataka za fajlove. Skladište treba da ima korisnike sa različitim privilegijama. Komponentu treba realizovati tako da bude odvojena specifikacija (API) u posebnu komponentu.

Pored specifikacije, potrebno je napraviti i dve implementacije ove specifikacije kao dve odvojene komponente. Prva implementacija skladišti fajlove na neko udaljeno skladište (na primer Google Drive, Dropbox, udaljeni računar u mreži) sa implementacijom autentifikacije, a druga treba da skladišti fajlove u neko lokalno skladište (lokalni fajl sistem). Obezbediti način da aplikacije koje koriste komponente mogu jednostavno da se prilagode za rad sa različitim implementacijama.

Neke od funkcionalnosti komponente koje je potrebno implementirati:

- inicijalizacija skladišta (može se implemenitrati kao kreiranje praznog foldera koji će biti korenski direktorijum skladišta)
- osmisliti da se komponentom podrži postojanje korisnika skladišta i napraviti konekciju na skladište za odgovarajućim nalogom, svaki korisnik treba da ima korisničko ime i lozinku koji se mogu čuvati u nekom struktuiranom fajlu u skladištu, takođe svaki korisnik ima i privilegije to su privilegija za snimanje fajlova, za preuzimanje i brisanje fajlova, svi korisnici mogu da pretražuju skladište. Kod implementacije udaljenog skladišta, ovi korisnici ne moraju da imaju veze sa formalnom autentifikacijom na Dropbox ili Google Drive.
- jedan korisnik kreira skladište i on ima sve privilegije i jedini on može da kreira ostale korisnike i da im dodeljuje privilegije
- pre nego što se pozove bilo koja operacija nad skladištem korisnik se konektuje na skladište i za svaku operaciju treba uključiti kontrolu privilegija, na kraju rada korisnik se diskonektuje
- kreiranje direktorijuma i praznih fajlova na određenoj putanji u skladištu (osmisliti razne načine zadavanja paternna kreiranja direktorijuma, kao u bash-u na primer `mkdir s{1..20}` kreira 20 direktorijuma pod imenom s1 do s20)

- smeštanje fajlova (jednog ili više) na određenoj putanji u skladištu sa ili bez metapodataka (putanja koja se prosleđuje treba da bude nezavisna od operativnog sistema, obraditi neke specifične situacije, na primer ako direktorijumi zadati putanjom ne postoje u skladištu)
- osmisлити način čuvanja metapodataka, vezu između fajla i metapodataka, obezbediti podršku sa proizvoljan skup metapodataka (dovoljno je podržati metapodatke u vidu parova ključ-vrednost, ali bez ograničenja šta mogu biti ključevi i vrednosti)
- upload kolekcije fajlova u vidu arhive (obezbediti operacije koje prihvataju arhive i operacije koje prihvataju listu fajlova pa ih same arhiviraju u zip arhivu, podržati da se ime arhive zada ili da se automatski generiše na osnovu nekih parametara, metapodataka)
- brisanje fajlova i direktorijuma iz skladišta
- podržati mogućnost da se zadaju ekstenzije fajlova koji se ne mogu skladištiti (na primer može da se definiše da skladište ne prihvata exe fajlove) i vraćanje greške ako neko pokuša da uploaduje fajl sa ekstenzijom koja se ne prihvata (ovo može da konfiguriše samo glavni korisnik)
- pregled sadržaja skladišta (podržati razne pretrage skladišta, na primer vrati sve nazive fajlove u direktorijumu, vrati nazive svih direktorijuma u nekom direktorijumu, vrati fajlove po imenu u direktorijumu i svim poddirektorijumima, vrati fajlove sa određenom ekstenzijom, vrati fajlove sa ili bez metapodataka i sl.)
- preuzimanje fajlova iz skladišta (zadaje se putanja koja može biti putanja do direktorijuma ili do fajla i odgovarajući element se preuzima iz skladišta)

Za sve tri komponente treba napisati dokumentaciju.

Pored komponenti potrebno je implementirati program koji će se pozivati preko komandne linije i koji će ilustrovati korišćenje komponente za skladište. Program se pokreće sa zadatom putanjom za korenski element skladišta, ukoliko na toj putanji ne postoji skladište treba izvršiti inicijalizaciju skladišta i kreirati jednog glavnog korisnika (unose se korisničko ime i lozinka preko komandne linije). Ukoliko postoji skladište na zadatoj putanji, unosi se korisničko ime i lozinka i vrši se konekcija na skladište. Zatim korisnik naredbama preko komandne linije poziva operacije nad skladištem (snimanje fajlova, skidanje fajlova, kreiranje direktorijuma,...). Glavni korisnik može da kreira druge korisnike sa određenim privilegijama.

Program treba da bude implementiran tako da radi sa specifikacijom komponente, i jednom implementacijom, ali da bude implementiran tako da poziva samo elemente specifikacije, tako da promenom implementacione komponente (zamene jednog vrsta skladišta drugim) koja podrazumeva promenu dependencija, podešavanje konfiguracije i ponovo pakovanje, program može jednostavno da se prilagodi za rad sa drugom vrstom skladišta.

Pakovanje komponenti i programa za komandnu liniju, specifikacija dependencija,

kao i generisanje dokumentacije treba da bude automatizovano nekim build alatom. Preporučeni jezik za implementaciju je Java, a alati za build Apache Maven ili Gradle.

Raspodela poena:

Dobro osmišljena specifikacija, kvalitet implementacije specifikacije - 10p

Implementacija dve komponente koje implementiraju specifikaciju -
 lokalno skladište - 5
 udaljeno skladište - 6

Dokumentacija za sva tri projekta- 1p

Implementacija programa za komandnu liniju - 5p

Automatizacija pakovanja biblioteka i izvršne aplikacije korišćenjem build alata - 3p

Napomene:

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.

Rok za predaju i odbrana:

Projekat se radi u timu od dvoje i rok za izradu je 8.11. u 12h (podne). Projekti se mogu poslati na mail bdimicsurla@raf.rs ili postaviti na neki privatni git repozitorijum i staviti profesorku kao kolaboratora (github ili bitbucket, nalog je [bdimicsurla](https://github.com/bdimicsurla)).

Deo studenata će projekat braniti na vežbama u 7. nedelji, a deo u kolokvijumskoj nedelji, raspored odbrane će biti objavljen na materijalima.

Članovi tima se posebno ocenjuju prema procenjenom uloženom trudu.