

## XML Tehnologije SIT – 04.07.2017. K1

=====

### Zadatak

Zadatak napraviti RESTful API, koji omogućava pružanje usluga rada više “online web servisa pošti”. Simulirati slanje pošiljke od prodavca ka kupcu, gde prodavac nakon dodavanja pošiljke, „šalje” prateći kod pošiljke kupcu, kako bi on mogao da prati trenutno stanje pošiljke. Za način interpretacije podataka koristiti JSON notaciju.

### Model - pojo

Entitet **pošta** poseduje vise **skladišta**. Svako **skladište** pripada tačno jednoj **pošti**. Svako **skladište** poseduje vise **paketa**, a svako **a svaki paket** pripada tačno jednom **skladištu**. Prilikom modelovanja klasa, **obezbediti sve attribute** kako bi se uspešno modelovao gore navedeni odnos.

#### Pošta

- ☐ Id (int)
- ☐ Skraćeni zapis (string)
- ☐ Naziv (string)
- ☐ Adresa (string)

#### Skladište

- ☐ Id (int)
- ☐ Skraćeni zapis (string)
- ☐ Adresa (string)
- ☐ Broj telefona (int)

#### Pošiljka

- ☐ Id (int)
- ☐ Kod\_za\_praćenje (string) – generiše se automatski kao (id paketa + skraćeni naziv pošte + masa pošiljke + skraćeni naziv skladišta)
- ☐ Datum prispeća (Date)
- ☐ Datum preuzimanja (Date)
- ☐ Masa (double)
- ☐ Stanje (string) – Može biti PRIMLJENO, U\_TRANZITU, ISPORUČENO

### Implementacija

Napraviti jedan glavni servis koji opslužuje sve pošte u sistemu. U njemu inicijalizovati 3 pošte, 1 u svakoj od njih po 3 skladišta, od po 10 paketa. Omogućiti **pristup**, **dodavanje**, **brisanje**, **izmenu** za svaki od gore navedenih entiteta služeći se REST specifikacijom, **obavezno** koristeći “resurs-podresurs (*nested resources*)” šablon. *Napomena: dodavanje i brisanje ne moraju da se implementiraju za entitet „Pošta”.*

*Svaka od akcija treba da bude logovana – zapisana u zapisnik. (koristiti java.util.Logger)*

## Uloge

U sistemu postoje 3 uloge:

- ☐ Admin
- ☐ Prodavac
- ☐ Kupac

Administratori imaju pristup svim servisima. On može da pretražuje sve pošiljke između dva datuma („**od**“ i „**do**“). Ukoliko prvi datum „**od**“ nije naveden tada se podrazumevaju sve pošiljke do „**do**“ datuma i obratno. Ukoliko nije naveden ni „**od**“ ni „**do**“ datum tada ispisati poruku o grešci da je neispravan zahtev (error code 500 – bad request). Datum se šalje URI-ju, gde je oblik datuma „**yyyy-mm-dd**“ npr. „**2017-04-28**“

Administratori imaju pristup svim servisima. Budući da administrator ima pristup svim pošiljkama, treba omogućiti **paginaciju** tako što će se kao parametri prosleđivati **broj stavki za prikaz, kao i broj strane**. Kao odgovor u telu zahteva dobijamo navedeni broj pošiljki (npr. imamo 18 pošiljki, tražimo 8 stavki za treću stranu, i dobijamo ispis preostale 2 pošiljke), u slučaju da je broj strane veći nego što ima pošiljki, u zaglavlju odgovora polje **Prekoračeno** staviti na vrednost 1, u suprotnom na 0.

Prodavac može samo da dodaje nove pošiljke. Kada prodavac **dodaje** novu pošiljku, u **zaglavlju odgovora** stiže povratna informacija **KodZaPraćenje** sa vrednosti koda za praćenje novo dodate pošiljke.

Tu informaciju će koristiti **kupac**, koji ima ulogu da pristupa pošiljkama isključivo na osnovu vrednosti **KodaZaPraćenje**, tako što će je uvrstiti u **zaglavlje zahteva** (*smatra se da je prodavac prosledio kupcu kod za praćenje pošiljke, tj. kupac ne može da vidi stanje pošiljke za koje nema kod*).

Kupac u zahtevu pored KodaZaPraćenje šalje i header **Registrovan**. Ukoliko u je vrednost Registrovan postavljena na 0, tada kupac nije u mogućnosti da izvrši pristup informacijama o bilo kom paketu.

Korisničko ime	lozinka
adminKorisnik	pass
prodavacKorisnik	pass
kupac1Korisnik	pass
kupac2Korisnik	pass

Implementirati DIGEST autentifikaciju. (koristiti kao pomoć *server/conf/conf-helper.txt* datoteku)

**Napomena:** kada se budu testirale funkcionalnosti samo **pristup** će biti testiran pomoću DIGEST autentifikaciju, ostale stavke će biti testirane bez sigurnosnih mehanizama (*Postman issue 932*).

## Postman

Alat koji će se koristiti za testiranje je Postman, gde ćete napraviti kolekciju za svaki od entiteta, i po jedan primer sačuvati HTTP zahtev. Krajnji rezultat eksportovati u datoteku.

Sačuvati vaš rad kao *br\_indexa.zip* na *C:/Temp* lokaciji.