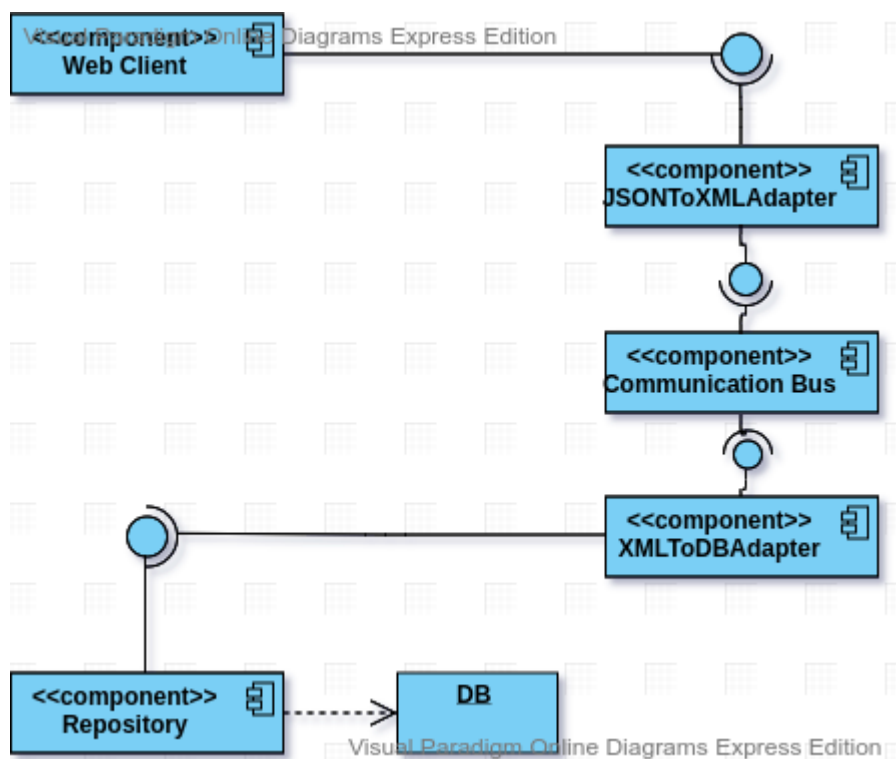


Projektni Zadatak P1 – Communication Bus

Potrebno je napraviti dizajn sistema, arhitekturu sistema, implementaciju i istestirati rešenje koje simulira rad i komunikaciju *Communication Bus* modula.



Način rada sistema

Web client komponenta dobija od strane svojih klijenata neki od zahteva:

GET, POST, PATCH, DELETE praćen odgovarajućom imenicom

Npr: GET /resurs/1. Zahtev se dobija u odgovarajućem JSON formatu:

```
{
  "verb": "GET",
  "noun": "/resurs/1",
  "query": "name='pera'; type=1",
  "fields": "id; name; surname"
}
```

Sama komponenta ima zadatak da proveri da li je zahtev dobro formatiran (da li prosleđeni glagol postoji, da li su vrednosti polja query i fields formatirana na odgovarajući način). Polja query i fields nisu obavezna, ona dodatno filtriraju resurse.

Kako Communication Bus ne razume JSON kao format podataka, već isključivo XML, neophodno je zahtev pretvoriti u XML i to je svrha JSONToXMLAdapter komponente. XML od prethodnog JSON zahteva bi bio :

```
<request>
  <verb>GET</verb>
  <noun>/resurs/1</noun>
  <query>name='pera';type=1</query>
  <fields>id; name; surname</fields>
</request>
```

Communication bus dobijeni zahtev samo prosleđuje svojoj adapter komponenti XMLToDBAdapter koja treba ovaj XML da pretvori u odgovarajući SQL upit (izgled upita će da zavisi od baze koju odaberete):

```
SELECT
  id,
  name,
  surname
FROM resurs
WHERE id = 1
AND name = 'pera'
AND surnatypeme = 1
```

Repository komponenta prima zahtev u ovom obliku i izvršava ga nad bazom podataka, dobija odgovarajući odgovor i dobijeni odgovor prosleđuje XMLToDBAdapter komponenti da ga pretvori u odgovarajući XML, svaki odgovor mora da poseduje polja: STATUS, STATUS_CODE i PAYLOAD. Dozvoljeni statusi su: REJECTED, BAD_FORMAT, SUCCESS, a status kodovi: 3000, 5000 i 2000 respektivno. Payload bi trebalo da sadrži dobijeni resurs odnosno u slučaju primeru XML polja id, name i surname u slučaju SUCCESS koda odnosno error-message polje sa porukom o grešci. Tako dobijeni XML Communication bus prosleđuje svojoj JSONToXMLAdapter komponenti koj ga transformiše u JSON. Zatim se odgovor prosleđuje Web Client komponenti koja ga vraća svojim klijentima. U slučaju da zahtev nije bio dobro formatiran Web Client odmah vraća BAD_FORMAT 5000 sa odgovarajućom porukom o grešci, ne prosleđuje zahtev dalje.

Napomena:

Zahtev pored polja navedenih u primeru može da poseduje i sledeća polja: connectedTo sa spiskom id-ova povezanih resursa(id=1; id=3), pri čemu je ovo dodatni filter, na primer ako postoji više resursa sa imenom Petar vrati mi onog koji je u vezi sa resursima Klavir i Odbojka, kao i connectedType gde se navode id-ovi odgovarajućih tipova, pri čemu je i ovo dodatni filter i koristan je kad na primer imam više resursa sa imenom Petar a potrebni su mi samo oni koji postoje u tipu veze roditelj. Kada klijent Web Client komponente kreira zahtev obavezan je da unese samo verb i noun, sve ostalo predstavlja samo dodatni filter koji može, a ne mora da unese.

Model podataka

Resurs:

- Id
- Naziv
- Ime
- Opis (u JSON formatu, ukoliko izabrana baza podržava JSON kao tip podatka kolone)
- Tip

Tip:

- Id
- Naziv

Veza:

- Id
- Id provog resursa
- Id drugog resursa
- Tip

Tip veze:

- Id
- Naziv

Kriterijum ocenjivanja

1. Dizajn i arhitektura rešenja
2. Korišćenje Scrum metodologije razvoja – definisanje User Story-a i taskova, planiranje i estimacija
3. Implementacija rešenja
4. CI ciklus
 - a. Build
 - b. UnitTestovi
 - c. Pokrivenost koda testovima