

| | |
|---|---|
| 1. Ülesande sisu lühidalt. Aruande struktuur ja formaat..... | 1 |
| Näide selle kohta kuidas võiks viidete osa teie dokumendis välja näha: | 2 |
| 2. Teemade nimekiri mille kohta tuleb otsida materjale ja kirjutada kokkuvõte..... | 3 |
| Teema 1: SOAP-teenuse veatöötlus. Vigade tagastamine SOAP-teenuse poolt SOAP-kliendile..... | 3 |
| Teema 2: SOAP Fault element..... | 4 |
| Teema 3: SOAP-teenus ja muud vead - võrgukatkestused ja SOAP-teenuse mittetöötamine. | 4 |
| 3. Lisapunktid..... | 5 |

1. Ülesande sisu lühidalt. Aruande struktuur ja formaat.

See ülesanne on põhimõtteliselt materjalide kogumise ja nende materjalide uurimise ülesanne. Materjaliotsingu tulemused tuleb viidetena panna selle ülesande dokumenti. Tuleb kirjutada oma sõnadega kokkuvõte milles proovite vastuseid leida selle ülesande teemade loetelus (vaata punkt 2.) toodud küsimustele.

Tehke dokument mis sisaldab kahte punkti:

1. Viited materjalidele teemade lõikes. (vähemalt 15)
2. Kokkuvõte teemast. (vähemalt 3 lk).

Otsida materjale punktis 2. toodud teemade kohta. Materjalideks sobivad artiklid, koodinäited, foorumite postitused, juhendid , tarkvara dokumentatsioon (sealhulgas API-de kirjeldused).

Materjale peaks olema vähemalt 15. Lisage dokumendi esimesse punkti („Viited materjalidele“) viidete nimekiri koos lühikese (3-4 lausest koosneva) selgitusega . Selgituses kirjutage lühidalt mis on materjali sisu, milles seisneb selle materjali kasulikkus. Mõned viited võivad viidata ka raamatutele mis ei ole otseselt internetist vabalt kättesaadavad kuid üldiselt on eeldatud et vähemalt 10 teie poolt viidatud materjali on ülesande kaitsmise ajal vabalt internetist kättesaadavad.

Kirjutage punkti 2 („Kokkuvõte teemast“) alla oma sõnadega kokkuvõte ülesandes käsitletud teemade kohta (vaata punktist 2. teemasid). Kokkuvõte peab olema vähemalt 3 lehekülje pikkune. Kokkuvõttes peaks sisalduma viiteid teie poolt kogutud materjalidele. Kokkuvõttes võib sisalduda materjalidest võetud koodinäiteid, skeeme, tsitaate ja muud infot (see on isegi soovitatav) kuid siin kehtib 2 tingimust:

- peab olema viide materjalile kust koodinäide on võetud
- materjalidest võetud infomatsiooni kokkuvõttes ei võeta kokkuvõtte pikkuse arvestamisel (mis peab olema vähemalt 3 lehekülge) arvesse.

Materjalid ja koodinäited mida leiate ja oma töös kasutate ei pea olema Java näited – sobivad materjalid mis räägivad suvalisel platvormil SOAP-teenustega tegelemisest.

Kui leiate materjale mis räägivad veebteenuste veahaldusest aga ei te ei ole kindel millise teema alla see materjal sobib siis lisage need materjalid esimese teemaga seotud materjalide alla („SOAP-teenuse veatöötlus. Vigade tagastamine SOAP-teenuse poolt SOAP-kliendile“).

Viited ei tohiks olla kõik ühte tüüpi (näiteks lingid foorumi postitustele). Kindlasti peaks olema viiteid ka artiklitele.

Näide selle kohta kuidas võiks viidete osa teie dokumendis välja näha:

-----viidete näide-----

Teema 1: SOAP-teenuse veatöötlus. Vigade tagastamine SOAP-teenuse poolt SOAP-kliendile

[1]. <http://stackoverflow.com/questions/1019141/should-web-services-throw-exceptions-or-result-objects>

Diskussioon selle kohta kas veebteenus peaks kliendile viskama vea või tagastama lihtsalt veakoodi. Teema algataja arvab et SOAP-kliendile vea viskamine ei ole päris hea idee. Kokku 6 vastust. Üks vastaja kirjutab „Read up on SOAP faults. This is the standard way to return arbitrary detail to a client“. Tundub et see lause on kasulik, siit selgub et SOAP fault on see andmestruktuur mille abil saab server kliendile edasi anda spetsiifilisemat veaga seotud infot (lisaks veateatele).

[2]. <http://www.javaworld.com/javaworld/jw-12-2006/jw-1220-wsexcept.html>

Artikkel. Siin räägitakse mõnedest parimatest praktikatest veebteenuste veatöötles. Tähelepanu pööratakse sellele kuidas teha serveri poolel vigade töötlemist ja veainfo tagastamist kliendile nii et see oleks „container-safe“ – mis tähendab et see vigade töötlemise ja tagastamise loogika oleks võimalikult vähe sõltub veebteenuste serveri (containeri) tarkvarast.

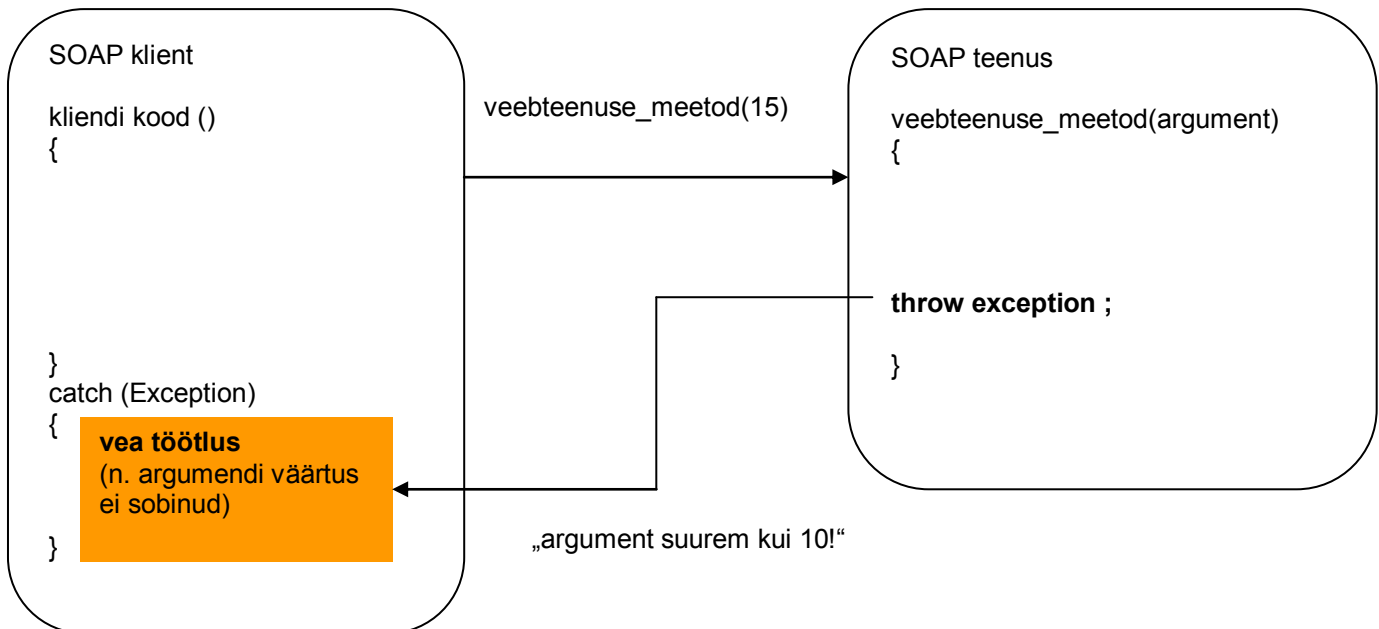
-----viidete näite lõpp-----

Pöörake materjalide otsimisel mõnevõrra tähelepanu ka materjali vanusele. Kuigi SOAP-tehnoloogia on vana tehnoloogia võiks materjalide vanus olla umbes aastast 2004 või uuem. Olulisi muudatusi SOAP-teenuste tehnoloogias ei ole viimasel ajal toimunud , sellepärast ei ole vaja ka seda et materjalid oleksid tingimata aastatest 2011 või 2010.

2. Teemade nimekiri mille kohta tuleb otsida materjale ja kirjutada kokkuvõte.

Teema 1: SOAP-teenuse veatöötlus. Vigade tagastamine SOAP-teenuse poolt SOAP-kliendile

SOAP-teenusel on vaja informeerida väljakutsujat (SOAP-klienti) et pöördumise teenindamisel tekkis viga. Vaja oleks teha nii et SOAP-teenuse väljakutsumisel selle vea korral käivituks SOAP-kliendi veatöötluse blokk.



Joonis 1. SOAP teenus ja viga serveril

Olukorra täpsustamiseks eeldame et meil on tegemist sellise situatsiooniga kus viga avastatakse serveri teenuse loogika poolt ja SOAP-kliendile on nüüd vaja sellega seoses tagastada (visata) viga. Näiteks võib võtta sellesama hinnakirja-teenuse koduülesandest 2.Sellel teenusel on üks argument. Veebteenus võib näiteks kontrollida kas sisendparameeter on numbriline. Võib kontrollida kas sisendparameeter jääb mingisse numbrite vahemikku (0....100). Võib kontrollida kas sisendparameetri väärtus on negatiivne või positiivne, ja nii edasi – vastavalt sellele millised on veebteenuse nõudmised korrektsele sisendparameetrile. Kui sisendparameetri väärtus ei ole veebteenusele sobiv siis tuleb väljakutsujale visata viga , näiteks („argument ei ole number!“). Veainfo tuleb tagastada väljakutsujale nii et väljakutsuja poolel jätkataks tämist veatöötluse blokist (catch-blokk käivitub).

Katsuge leida vastused kõigile (soovitavalt) või vähemalt osale järgmistest küsimustest:

- Kuidas on võimalik SOAP serveri ja SOAP kliendi puhul nii teha et SOAP server viskab vea (throws exception) ja SOAP klient püüab selle oma veatötluse blokis (catch) kinni?
- Mida tuleb serveri poolel vea viskamiseks KLIENDILE teha , kuidas programmeerida? Koodinäiteid kui leiate.
- Mida tuleb kliendi poolel teha, kuidas SOAP teenuse poolt visatud viga kätte saada?
- Kuidas saada kätte SOAP teenuse poolt visatud veateadet?
- Kuidas saada kliendi poolel teada SOAP teenuse poolt saadetud veatüüpi?
- Kuidas saada kliendi poolel teada SOAP teenuse vea koodi?

Teema 2: SOAP Fault element.

- Mis on „SOAP Fault element“?
- Kuidas on see SOAP Fault element seotud antud ülesande teemaga?
- Milline on SOAP Fault element-i struktuur?
- Mis eesmärgil kasutatakse alamelementi „<detail>“ selles SOAP Fault elemendi struktuuris ? Kuidas on see „<detail>“ alamelement seotud SOAP-kliendiga?

Teema 3: SOAP-teenus ja muud vead - võrgukatkestused ja SOAP-teenuse mittetöötamine.

Teemas 1 oli juttu situatsioonist kus SOAP teenus töötab aga talle ei sobi kliendi poolt saadetud andmed (SOAP-meetodi argumentid) ja ta tagastab kliendile vea.

- Kuidas klient püüab kinni ja käsitleb situatsiooni kus SOAP-teenus ei ole kättesaadav (ehk siis server on „maas“? Soovitavad koodinäited materjalidest.
- Kuidas klient püüab kinni ja käsitleb situatsiooni kus kliendil puudub võrguühendus?
- Kuidas klient püüab kinni ja käsitleb situatsiooni kus serveri vastust tuleb liiga pikalt oodata (ja võib-olla vastust ei tulegi)?

3. Lisapunktid.

Lisapunkte on võimalik saada programmeerimise eest. Tuleks täiendada koduülesande 2 lahendust lisades sinna ühe veasituatsiooni käsitlemine . Hinnakirja teenusele tuleks kliendi poolt edasi anda argument ja serveri pool peaks selle argumendi väärtust mingi loogika alusel kontrollima. Näiteks võiks kontrollida kas parameeter on number. Kui sisendi väärtus nõutud reeglile ei vasta siis tuleks kliendile visata viga. Kliendipoolne töö peaks nüüd edasi minema kliendi veatöötuse blokest . Veatöötuse osas (catch-blokis) võib kirjutada lihtsalt serveri poolt saadetud veainfo konsoolile, muud pole seal vaja teha.