

Ülesanne 5. Äriprotsess võrgus. IDU0080

1. Sissejuhatus. Materjalid ja ülesande sisu.....	1
2. Lahenduse skeem ja kommentaarid üksikute protsessi sammude kohta.....	4
2.1. Tellimuste päring id järgi tund.ttu.ee teenuselt.	7
2.2. Teenuste rakenduse kullerfirmade teenus.....	8
2.3. Teenuste rakenduse pakkumiste teenus.	9
2.4. Millised võiksid/peaksid olema ise-tehtud teenuste meetodid : argumendid ja tagastatavad tüübid.....	10
2.5. veatöötusest – pole vaja.....	11
3. Lahenduse arhitektuur.....	12
3.1. Kuidas teha nii et äriprotsess oleks käivitava Eclipsest või käsurealt ja oleks samal ajal ka veebteenusena kasutatav?	13
4. Täiendavad nõuded.....	15
4.1. logifailide kirjutamine äriprotsessi ja teenuste rakenduse poolt.....	15
5. Kuidas võiks välja näha töötava rakendus koduarvutis ja selle töötava rakenduse käivitamine ja töö	18
5.1. Lahenduse seadistamine ja käivitamine	19

1. Sissejuhatus. Materjalid ja ülesande sisu.

Otselink materjalide kataloogile:

<http://maurus.ttu.ee/idu0080/5/>

Näide selle kohta kuidas võiks ülesande tegemist alustada ja kuidas kasutada tund.ttu.ee teenuseid

ProductServiceClientServiceClient_naide.zip

Viited olemasolevatele teenustele mida peab ülesandes kasutama:

<http://tund.ttu.ee/orderservice/orders>

tund.ttu.ee veebteenuste andmebaasi struktuur ja testandmed

<http://maurus.ttu.ee/idu0080/5/db/>

(tegemist on PostgreSQL-i andmebaasiga, soovi korral on võimalik selles kataloogis olevate skriptide abil endale PostgreSQL-i andmebaas teha et paremini vaadata mis andmed teenuste taga on)

NB! Kõikide teenuste , klasside, meetodite ja rakenduse nimed, mis viitavad selle ülesande lahenduse sellele osale mida peate ise tegema, on vabalt valitavad. Kui selles juhendis nendele ongi mingid nimed antud joonistel ja koodinäidetes siis need nimed ei ole teile kohustuslikud vaid on toodud siin ära ainult selleks et kuidagi peab nendele viitama.

Ülesande sisu, protsessi sammud:

Ülesande sisuks on realiseerida Java rakendusena üks äriprotsess mis tegeleb mingi kaupade tellimusele parima transporditeenuse valiku ja tellimisega.

1. Äriprotsessi sisendiks on tellimuse id mille järgi tund.ttu.ee veebteenuse kaudu päritakse tellimuse andmed.

Tellimuse andmed sisaldavad tellimuse maksumust, müüja aadresse (neid võib olla mitu) ja kohaletoimetamise aadressi (ainult üks).

2. Kui tellimuse andmed on käes, küsitakse enda kirjutatud teenuselt (see teenus on eraldi rakenduses) kullerfirmade nimekiri. Teenuste rakendus omakorda küsib selle kullerfirmade nimekirja teiselt tund.ttu.ee teenusel. Äriprotsessile saadetakse vastuseks objektide massiiv mis sisaldab kullerfirmasid (kullerfirma id ja nimi).

3. Äriprotsessi poolt kätte saadud kullerfirmade nimikiri käiakse tsükliga läbi ja iga kullerfirma kohta saadetakse eraldi SOAP päring transporditeenuse pakkumise saamiseks. Teises (nn. teenuste) rakenduses olev SOAP-teenus vastab nendele pakkumiseküsimise päringutele ja saadab vastuseks pakkumise andmed: transporditeenuse hinna ja ligikaudse kohaletoimetamise aja.

Transporditeenuse hind on funktsioon :

- kullerfirma andmetest (saadakse tund.ttu.ee teenuselt meetodi **getCourierById()** väljakutsumisel. Kullerfirma andmed sisaldavad kullerfirma aadresse (neid võib olla mitu) ja protsenti tellimuse hinnast („percent_from_order”). Transporditeenuse maksumus ongi see protsent tellimuse hinnast mis on kullerfirma andmetes „percent_from_order” atribuudis.
- Tellimuse andmetest. Tellimuse andmed sisaldavad tellimuse hinda (transpordi maksumus on kullerfirma protsent tellimuse hinnast) ja tellimuse müüja aadresse (selle.addresses) ning tellimuse kohaletoimetamise aadressi (shipping address).
Pakkumise hinna arvutamisel peavad kehtima järgmised reeglid:
 1. kui tellimuse kohaletoimetamise aadressi maakond (**county**) langeb kokku vähemalt ühe kullerfirma aadressi maakonnaga (county) siis on transpordi hind **30%** odavam.
 2. kui vähemalt üks müüja aadressi maakond (**county**) langeb kokku mõne kullerfirma aadressi maakonnaga (**county**) (selle kullerfirma aadresside hulgast mille pakkumist parajasti arvutatakse) siis on transpordi hind **30%** odavam.

Nii et kokku võib aadresside maakonna-väljade kokkulangevuste korral olla transpordi hind kuni **60%** odavam.

Kohaletoimetamise päevade arv arvutatakse pakkumise tegemise meetodis randomiga , juhuslik arv vahemikus 1..7

Lisaks hinnale ja kohaletoimetamise päevade arvule genereeritakse pakkumisele ka mingi kood (pakkumise id, „offer id”) mis äriprotsessile tagasi saadetakse. Kood on suvaline kombinatsioon tähtedest ja numbritest „uppercase”-is. Näiteks: **RA4RCKD61K**

4. Transporditeenuse pakkumised kogutakse vastuvõtja (äriprotsessi poolel) massiivi ja arvutatakse igale pakkumisele pakkumise headus. Valemi võib iga rühm ise välja mõelda kui tahab aga võib kasutada ka siin etteantud valemit:

Pakkumise headus = transpordi hind * 0.01 * kohaletoimetamise päevade arv

Parim pakkumine on see millel on „headust” väljendav arv kõige väiksem.

Kui kasutate mingit oma valemit pakkumiste headuse arvutamiseks siis arvestage et selles valemis tuleb kindlasti mingil moel arvesse võtta mõlemaid pakkumise parameetreid – transporditeenuse hinda ja kohaletoimetamise aeg.

Äriprotsess saab nüüd pakkumised headuse alusel järjestada ja valida välja kõige soodsama transporditeenuse pakkumise.

5. Sobiv pakkumine on nüüd välja valitud ja saab teha pöördumise kolmanda ise-tehtud teenuse poole (teenuste rakenduses) – saab väljavalitud transporditeenuse pakkumise alusel teenuse ära tellida. Teenuse tellimiseks kutsutakse välja transporditeenuse tellimise teenus (alumisel joonisel nimega **TransportOrderService**) ja antakse parameetrina edasi ainult väljavalitud pakkumise id (üks string). Transporditeenuse tellimise teenus saadab tagasi saadetise nn. „**tracking number**”-i (selle järgi saavad kliendid tavaliselt jälgida saadetise liikumist transpordisüsteemides).

„Tracking number” genereeritakse järgmise reegli alusel:

Väikesteks tähtedeks (lowercase) teisendatud pakkumise number + „#” + jooksev kuupäev

Näiteks:

Kui pakkumise id on: **RA4RCKD61K**

Siis tracking number on: **ra4rckd61k#20110424**

6. Nüüd on olemas kõik andmed selleks et protsessi lõpp tund.ttu.ee teenuse abil registreerida:

- on olemas tellimise id (order id)
- väljavalitud pakkumise hind ja kohaletoimetamise (umbkaudne) aeg ja pakkumise teinud kullerfirma nimi
- saadetise „tracking number” mis saadi transporditeenuse tellimise tulemusena.
- rühma nimi (mis võib olla konstandina äriprotsessi rakendusse sisse kirjutatud)

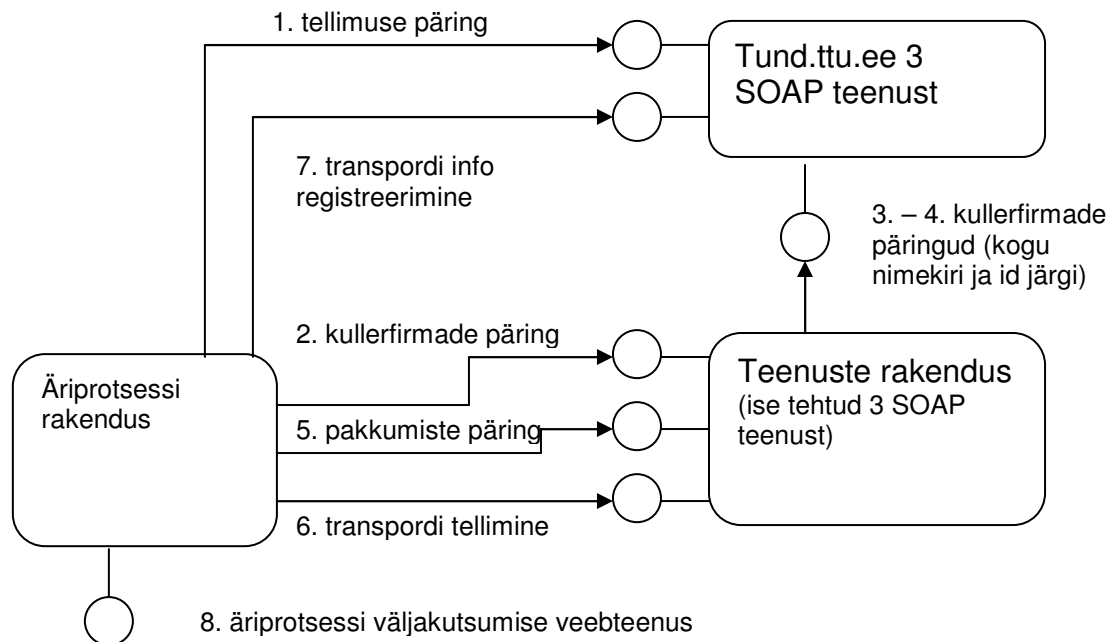
Nüüd tuleks välja kutsuda tund.ttu.ee teenus OrderShipment ja antud kaubatellimuse jaoks ja väljavalitud ja tellitud transporditeenus registreerida, kutsudes välja meetodid **insertOrderShipment()**;

lõik insertOrderShipment()-meetodi serveripoolsest koodist:

```
insertOrderShipment(@WebParam(name="order_id") int order_id,@WebParam(name="ryhm_name") String ryhm_name,@WebParam(name="tracking_number") String tracking_number,@WebParam(name="courier_name") String courier_name,@WebParam(name="approx_delivery_date") Date approx_delivery_date,@WebParam(name="shipping_price") double shipping_price)
```

Viimase tund.ttu.ee teenuse väljakutsumise tulemusena sisestatakse kirje tund.ttu.ee PostgreSQL-i andmebaasi tabelisse, selle tabeli sisu ehk

äriprotsessi tulemusi on võimalik vaadata SoapUI abil tund.ttu.ee serverist , meetod nimeks **getOrderShipmentsByOrderIdRyhmnName()**.



Nagu skeemi pealt näha peab äriprotsess ise ka olema välja kutsutav veebteenusena, selle veebteenuse äriprotsessi käivitamise meetodile antakse argumendina ette tellimuse id („order_id”)

2. Lahenduse skeem ja kommentaarid üksikute protsessi sammude kohta.

Põhimõtteliselt on tegemist süsteemiga mis koosneb 7-st SOAP-teenuse kliendist ja 6-st SOAP-teenusest.

„kliendi” all ei ole siin mõeldud mitte eraldi rakendust vaid mingit osa rakendusest, mingeid klasse ja meetodeid mis tegelevad otseselt mingite SOAP-teenustega suhtlemisega.

Lahenduses tuleb teha kokku **7 klienti**:

1. tellimuse päring tund.ttu.ee **OrderService** teenuse käest.
2. kullerfirmade päring enda kirjutatud teenuse käest
3. kullerfirmade päring tund.ttu.ee **CourierService** käest mida kasutab enda tehtud kullerfirmade teenus
4. kullerfirmade pakkumiste päring endatehtud teenusele mis genereerib transpordi-pakkumisi
5. ühe kullerfirma andmete päring tund.ttu.ee **CourierService** teenusele mis pärib kullerfima pakkumise tegemiseks kullerfima andmed firma id järgi (vaja on

kullerfirma aadresse ja teenuse maksumuse protsenti tellimuse hinnast selleks et transpordi-hind arvutada)

6. transporditeenuse tellimuse pöördumine (võitnud pakkumise alusel) endatehtud transporditeenuse tellimuse teenuse poole.
7. protsessi tulemuste registreerimine tund.ttu.ee **OrderShipmentService** teenuses.

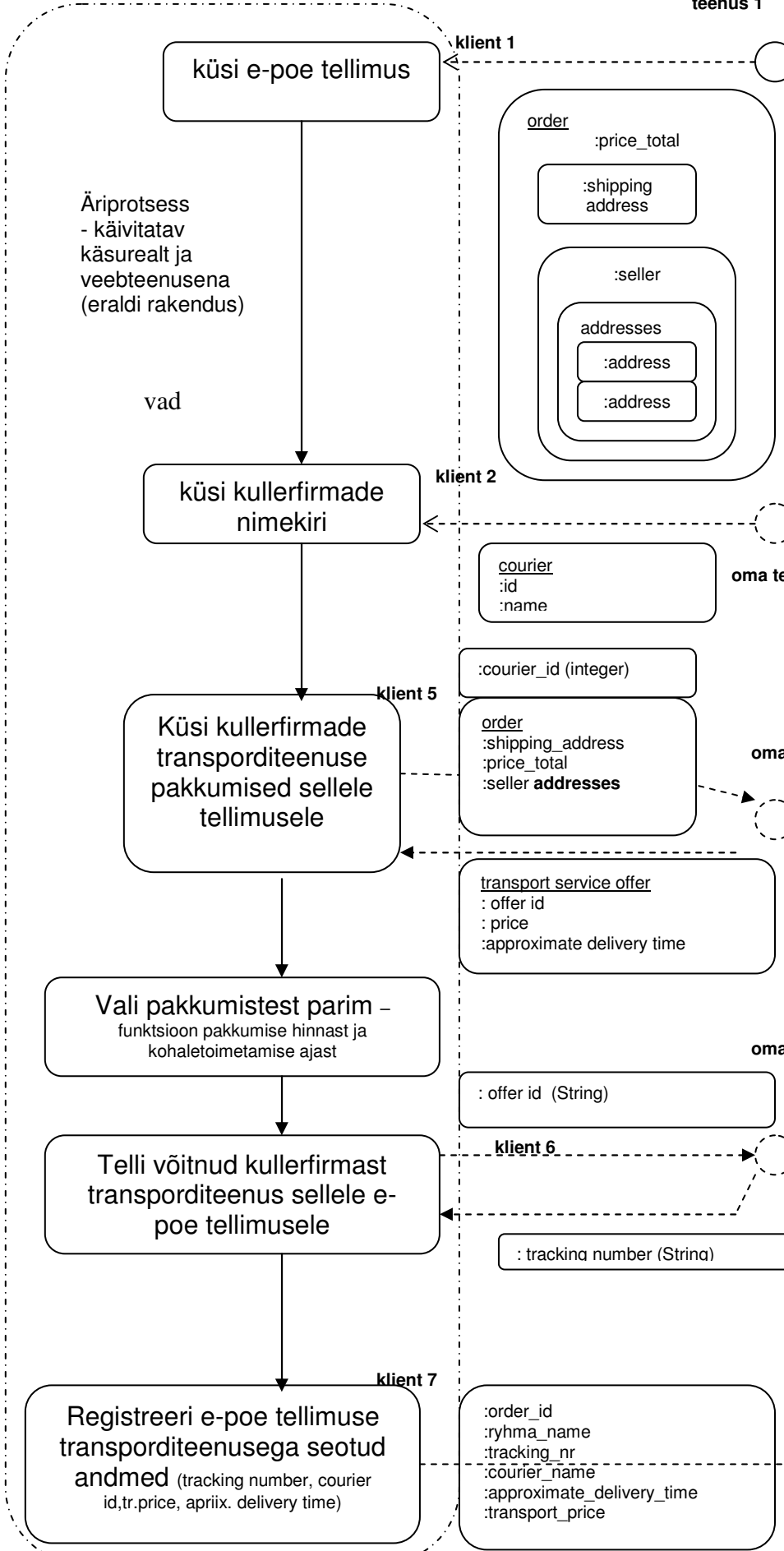
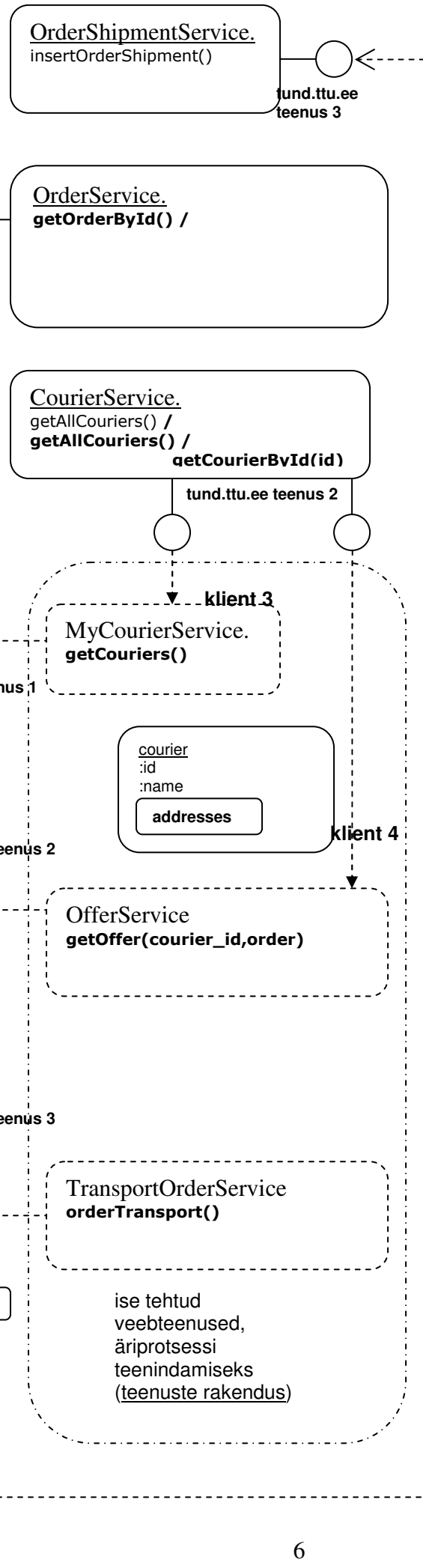
SOAP-teenustest **3 teenust** on juba valmis ja töötavad tund.ttu.ee serveril:

1. tellimuste päringu teenus (**OrderService**)
2. kullerfirmade päringu teenus (**CourierService**)
3. tellitud transporditeenuse registreerimise teenus (**OrderShipmentService**)

4 SOAP teenust tuleb enda kirjutada , need teenused peaksid olema eraldi "teenuste" rakenduses (s.h. eraldi Eclipse projektina)

1. Kullerfirmade nimekirja teenus (kasutab andmete saamiseks tund.ttu.ee vastavat **CourierService** teenust), sisaldab seega ka ühte SOAP-teenuse klienti.
2. pakkumiste tegemise teenus (kasutab andmete saamiseks tund.ttu.ee vastavat **CourierService** teenust), sisaldab seega ka ühte SOAP-teenuse klienti.
3. Transporditeenuse registreerimise teenus, väljastab tellimuse (saadetise) „tracking numbrit“

olemasolevad
veebteenused
serveril
tund.ttu.ee



Äriprotsess
- käivitav
käsurealt ja
veebteenusena
(eraldi rakendus)

vad

ise tehtud
veebteenused,
äriprotsessi
teenindamiseks
(teenuste rakendus)

Kullerfirma pakkumiste tegemise reegel:

Pakkumise hind = protsent tellimuse hinnast . Protsent sõltub kullerfirmast, serverilt tund.ttu.ee saadud kullerfirmade andmetes sisaldub ka atribuut „percent_from_order” kus on kirjas kui suure protsendi tellimuse maksmusest selle kullerfirma transpordi hind moodustab.

Kui tellimuses mõni müüja aadresside maakondadest langeb kokku mõne kullerfirma aadressi maakonnaga siis on pakkumise hind 30% madalam.

Kui kohaletoimetamise aadressi maakond langeb kokku mõne kullerfirma aadressi maakonnaga siis on pakkumise hind 30% madalam.

Kohaletoimetamise pakkumise reegel:

Pakkumise headus = transpordi hind * 0.01 * kohaletoimetamise päevade arv

Valida **min**(kohaletoimetamise pakkumise headus). (parim on see pakkumine millel on pakkumise headuse number (rank) kõige väiksem.

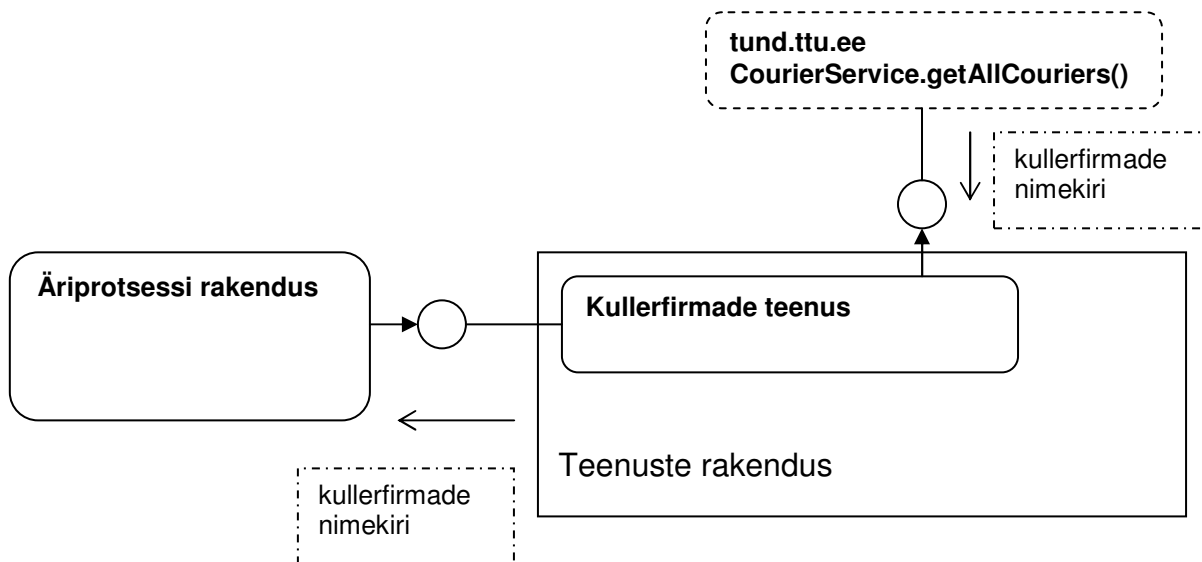
Pakkumiste headuse arvutamise valemit võib ise muuta ka, võib mingit muud valemit rakendada – oluline on et see valem võtaks arvesse nii transporditeenuse hinda kui kohaletoimetamise aega (päevade arvu).

2.1. Tellimuste päring id järgi tund.ttu.ee teenuselt.

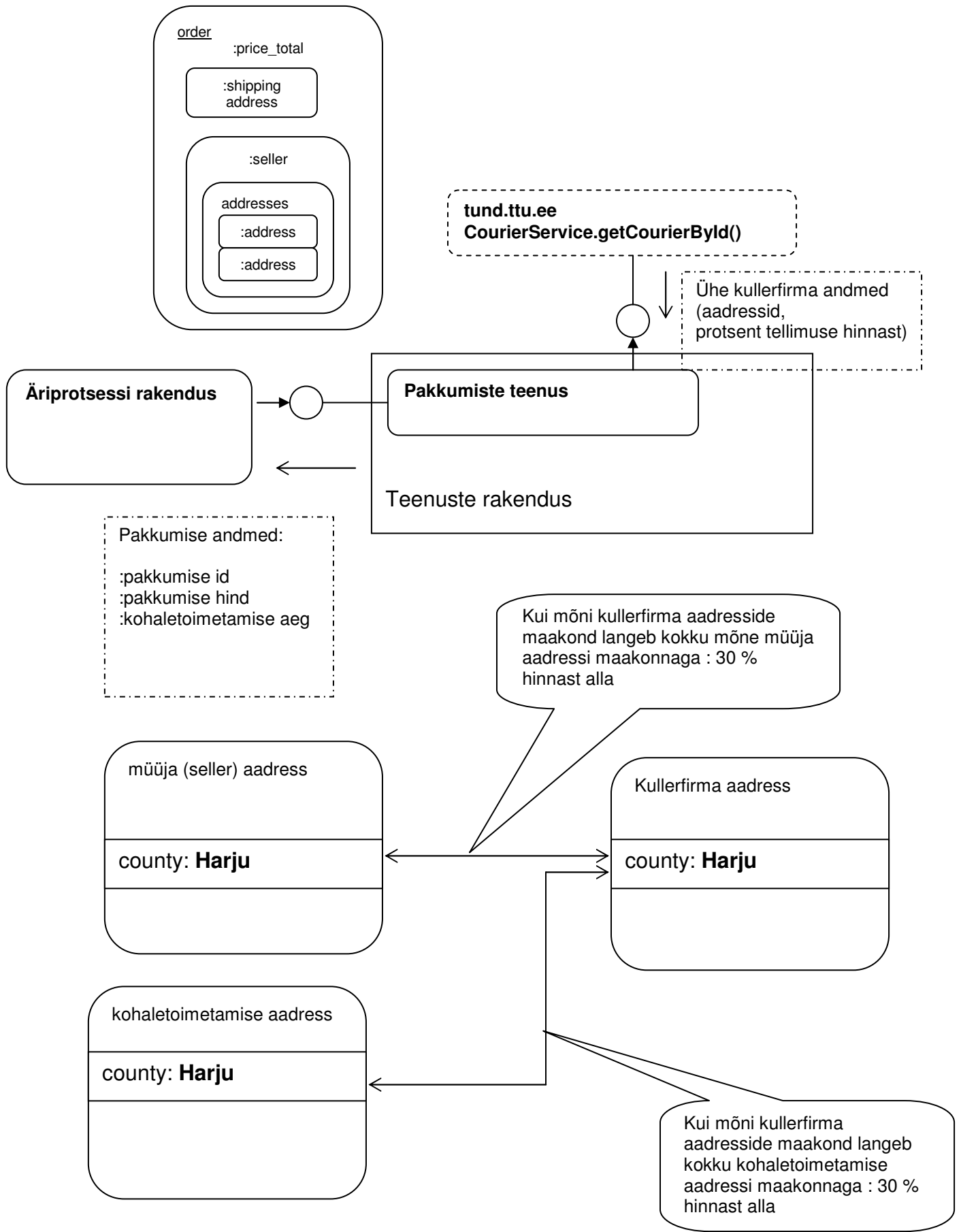
Äriprotsessi alustamiseks peab ette andma mingi tellimuse id („order_id”) mille järgi tellimuse andmed tund.ttu.ee teenuselt küsitakse. Tellimusi on tund.ttu.ee andmebaasis vähemalt 21 ja järjenumbrid algavad 1-st, seega võib päritava tellimuse id-ks olla maksimaalselt 21.

2.2. Teenuste rakenduse kullerfirmade teenus.

Ise tehtud kullerfirmade teenus on tegelikult nagu proxy-teenus mis küsib omakorda kullerfirmade andmed tund.ttu.ee teenuselt CourierService ja saadab päringule vastuseks kullerfirmade objektide massiivi.



2.3. Teenuste rakenduse pakkumiste teenus.



Transpordi-pakkumiste hinna arvutamisel saadakse sisendiks tellimus millele hakatakse pakkumist arvutama. (Tellimuse andmed saadakse protsessi kõige esimese veebteenuse väljakutse tulemusena kus küsitakse tund.ttu.ee teenuselt „order_id” järgi ühe tellimuse andmed. Tellimus sisaldab müüja objekti („seller”), müüja objekt sisaldab aadresse (1..n) , „addresses”.

Pakkumise arvutamisel tuleb vaadata läbi kõik müüja aadressid ja kui vähemalt üks nende aadresside maakondadest (atribuut ”county”) langeb kokku kullerfirma ühega kullerfirma aadresside maakondadest (võib ka rohkem kokkulangevusi olla aga see meid ei huvita, tähtis on et „vähemalt üks” langeb kokku) – siis läheb pakkumise hind 30 % odavamaks.

Kui vähemalt üks kullerfirma aadressidest (county) langeb kokku tellimuse kohaletoimetamise aadressiga (county) – siis läheb pakkumise hind veel 30% odavamaks.

Väljavõtte töötava lahenduse koodist:

```
-----  
int offer_discount_pc = 0;  
double offer_discount = 0;  
  
tr_price_total = (double) ((order.getPrice_total()*percent_from_order)/100) - ;  
  
if (IsMatchWithSellerAddresses(courier.getAddresses(), order.getSeller().getAddresses()))  
{  
    offer_discount_pc = offer_discount_pc + 30 ;  
}  
  
if (IsMatchWithShippingAddress(courier.getAddresses(), order.getShipping_address()))  
{  
    offer_discount_pc = offer_discount_pc + 30 ;  
}  
  
offer_discount = (double) ( tr_price_total*offer_discount_pc)/100;  
tr_price_total = tr_price_total - offer_discount ;  
-----
```

2.4. Millised võiksid/peaksid olema ise-tehtud teenuste meetodid : argumendid ja tagastatavad tüübid.

Millised võiksid olla ise-tehtud SOAP-teenuste meetodid ja neile sisendiks antavad tüübid ja tagastatavad tüübid – selle kohta võib saada mõtteid kui panete enda arvutisse üles töötavad näiterakendused ja vaatate teenuste rakenduse veebteenuste wsdl-e ja uurite neid teenuseid SoapUI-ga – vaata punktist 5 „**Töötava lahenduse näide**”

Kui olete näite oma arvutisse üles pannud siis näete lahenduste teenuste osa wsdl-e järgmisel aadressil:

<http://localhost:8080/5services/services/>

Available SOAP services:

CourierList <ul style="list-style-type: none"> • getCourierList 	Endpoint address: http://localhost:8080/5services/services/CourierServicePort WSDL : http://service.courierservice.idu0080.ttu/ CourierServiceService Target namespace: http://service.courierservice.idu0080.ttu/
OrderService <ul style="list-style-type: none"> • order 	Endpoint address: http://localhost:8080/5services/services/TransportOrderServiceImplPort WSDL : http://transportorderservice.idu0080.ttu/ TransportOrderServiceImplService Target namespace: http://transportorderservice.idu0080.ttu/
TransportOfferService <ul style="list-style-type: none"> • getOffer 	Endpoint address: http://localhost:8080/5services/services/OfferServicePort WSDL : http://service.offerservice.idu0080.ttu/ OfferServiceService Target namespace: http://service.offerservice.idu0080.ttu/

2.5. veatötlusest – pole vaja.

Veatötlusega pole selles ülesandes vaja tegeleda, eeldame et kõik andmed mis liiguvad on korrektsed ja erindeid ei teki.

3. Lahenduse arhitektuur.

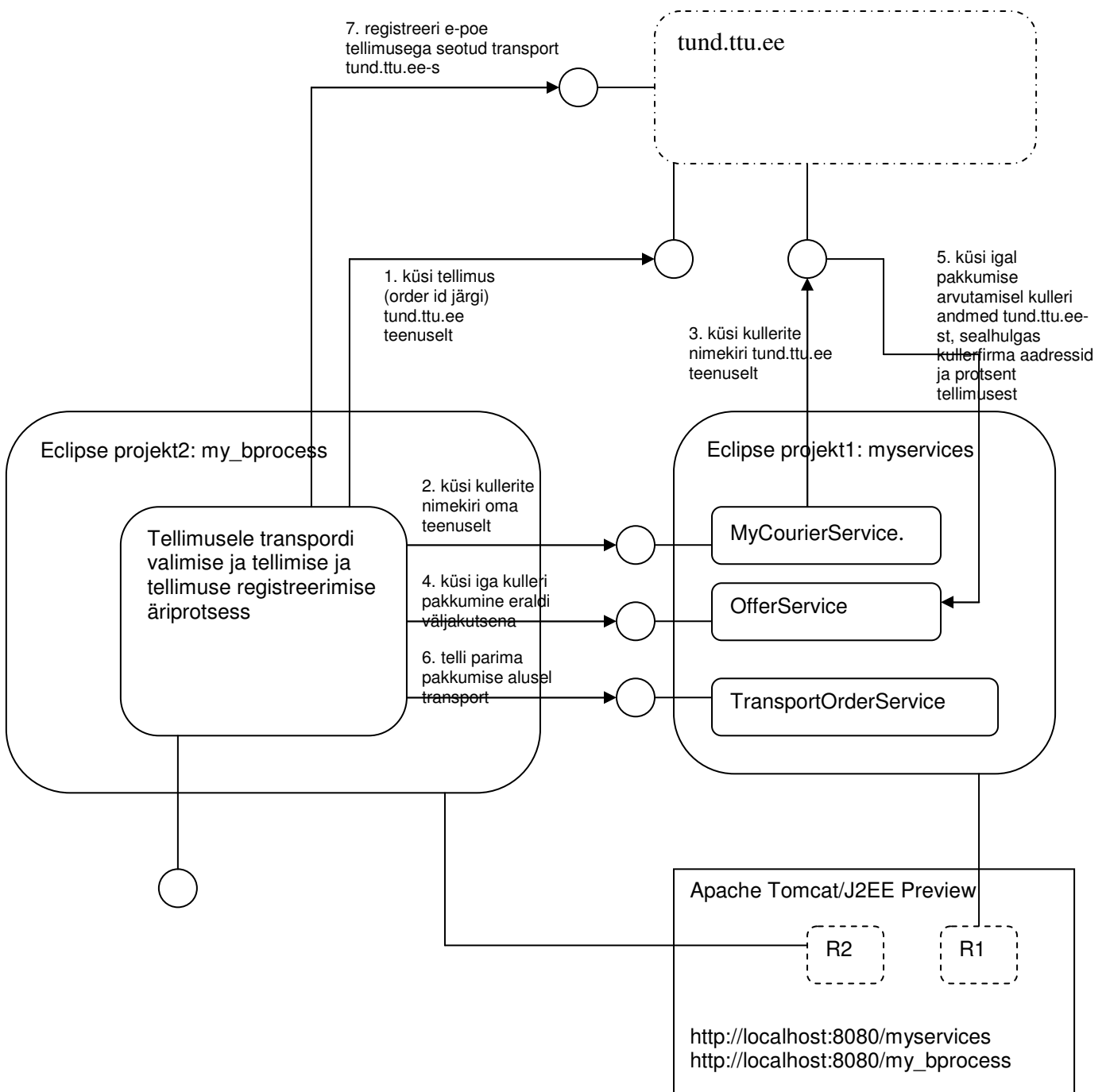
Lahendus võiks koosneda kahest Apache tomcat-i all töötavast veebirakendusest:

1) teenuste rakendus mis sisaldab kolme selle ülesande jaoks vajalikku teenust mis tuleb ise teha. Osa teenuseid kasutab omakorda teenuseid tund.ttu.ee serverilt.

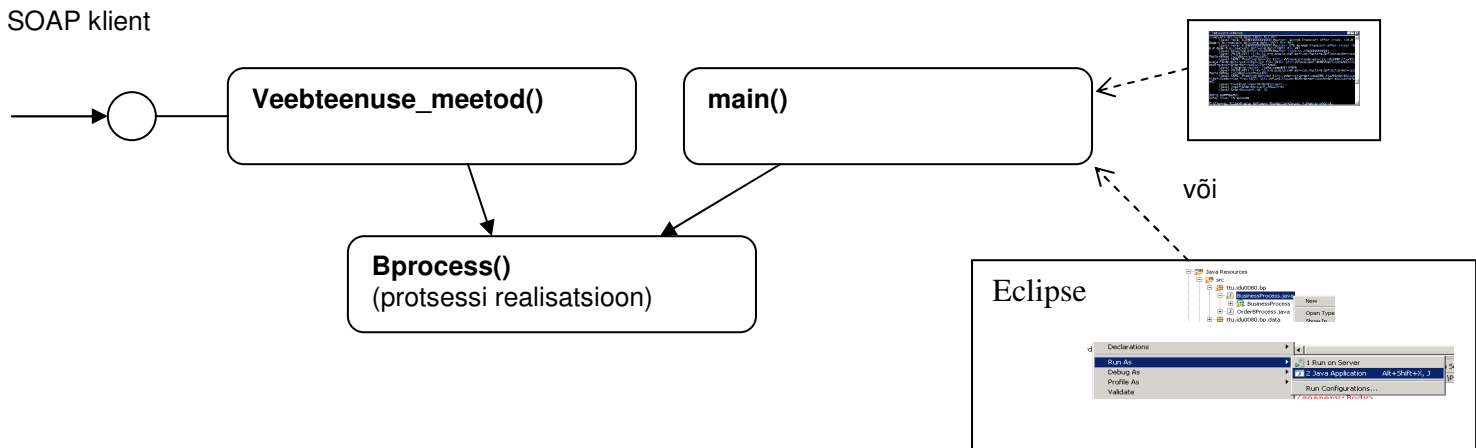
2) äriprotsessi rakendus mis sisaldab teenuste väljakutseid ja on ka ise välja kutsutav veebteenusena.

Äriprotsessi peab olema võimalik käivitada ka käsurealt (vaata töötavat lahendust).

Mõlemaid rakendusi peaks olema võimalik käivitada Eclipsest ja nad võiksid töötada Apache Tomcat-i keskkonnas (server runtimeks sobib ka J2EE Preview, töötav näide on tehtud Tomcat-ile)



3.1. Kuidas teha nii et äriprotsess oleks käivitava Eclipsest või käsurealt ja oleks samal ajal ka veebteenusena kasutatav?



Võib teha näiteks nii:

1. lisada (mingi suvalise nimega) äriprotsessi klassile meetod mis kutsub välja tegelikku äriprotsessi realisatsiooni. Selle meetodist saab teha veebteenuse (Eclipse wizardiga).
2. lisada äriprotsessi klassile main-meetod mida saab käivitada Eclipsest või käsurealt ja mis ka kutsub välja äriprotsessi realiseerivat meetodit. Alumises koodinäites on äriprotsessi realiseeriva meetodi nimeks **getTransportOfferToOrder()** – see meetod sisaldab äriprotsessi kõiki samme.

Kui „**Veebteenuse_meetod()**” nüüd veebteenuseks teha (Eclipse wizardi abil) ja rakendus nüüd (eeldusel et rakendus on loodud „**Dynamic Web Project**”-ina) serverile üles seada siis on „**Veebteenuse_meetod()**” kättesaadav üle SOAP-protokoll.

Eclipsest on aga äriprotsessi ka võimalik käivitada, Eclipsest käivitamise puhul on võimalik otse lähtekoodis redigeerida etteantavat parameetrit „**order_id**”.

Selles ülesandes piisab selles et äriprotsess oleks käivitav Eclipse alt. Tomcat-i serverile war-failina kopeeritud rakenduse protsessi on võimalik käivitada ka käsurealt ant-i abil, nii on tehtud töötava rakenduse näites (vaata järgmist punkti) aga selle ülesande tegijatel pole seda vaja teha – piisab kui on võimalik Eclipsest käivitada.

```
@WebService(targetNamespace = "http://bp.idu0080.ttu/", endpointInterface =
"ttu.idu0080.bp.OrderBProcess", portName = "BusinessProcessPort", serviceName =
"BusinessProcessService")
```

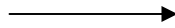
```
public class BusinessProcess implements OrderBProcess {
    private List<ttu.idu0080.bp.data.BPTransportOffer> BusinessProcessTransportOffers =
null;
    private ttu.idu0080.bp.data.BPTransportOffer SelectedTransportOffer = null;
    private int order_id = 0;
    private String tracking_number = "";
    private String ryhmName="RYHM34";
    private int order_shipment_id = 0;
    static final long DAY_MS = 1000 * 60 * 60 * 24;
```

```
public static void main(String[] args) throws Exception {
    writeToFile("protsessi algus", true);
    writeToFile("-----", false);

    int order_id = 0;
    if (args.length == 0)
    {
        order_id = 1;
    }
    else
    {
        order_id = Integer.parseInt(args[0]);
    }
    System.out.println("ORDER ID:" + order_id);
    BusinessProcess bp = new BusinessProcess();
    bp.setOrderId(order_id);
    bp.getTransportOfferToOrder();
    writeToFile("protsessi lopp", true);
    writeToFile("-----", false);
}
```

seda
käivitame
Eclipsest või
käsurealt)

tegelik protsess on selles meetodis



```
public void RunOrderShipmentBusinessProcess(int order_id)
{
    BusinessProcess bp = new BusinessProcess();
    bp.setOrderId(order_id);

    try {
        writeToFile("protsessi algus", true);
        writeToFile("-----", false);
        bp.getTransportOfferToOrder();
        writeToFile("protsessi lopp", true);
        writeToFile("-----", false);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

sellest meetodi
teeme
veebteenuseks

tegelik protsess on selles meetodis

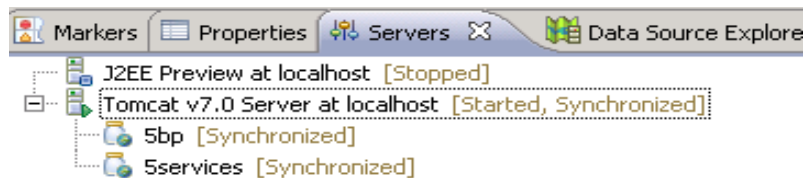


```
-----build.xml-----
<target name="run.process" depends="init">
    <java classname="ttu.idu0080.bp.BusinessProcess" classpathref="lib.classpath">
        <arg value="${order_id}" />
    </java>
</target>
```

Käivitamine käsurealt kui rakendused on kopeeritud Tomcati webapps kataloogi ja seal installeeritud (nagu töötava näite puhul järgmises punktis).

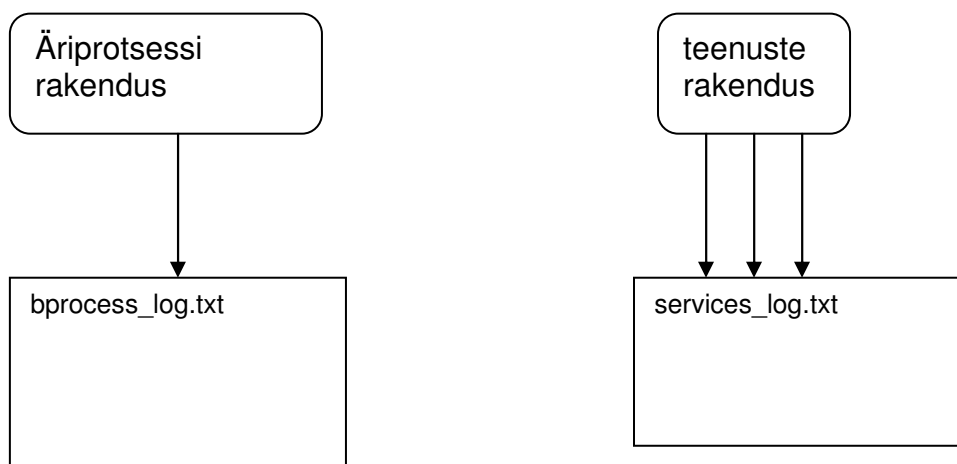
```
cd C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\5bp
ant -Dorder_id=14 run.process
```

Juhul kui mõlemad rakendused (oma teenused ja äriprotsess) on Eclipsest serverirakendustena käivitatud siis näeks Eclipse serverite aken välja umbes nii:



4. Täiendavad nõuded.

4.1. logifailide kirjutamine äriprotsessi ja teenuste rakenduse poolt.



Äriprotsessi rakendus ja teenuste rakendus peaksid tekitama mõlemad oma logifaili kuhu pannakse kirja olulisem info tegevuste ja andmete kohta.

Logifailide näited asuvad kataloogis:

http://maurus.ttu.ee/idu0080/5/lahendus/services_log_example.txt

http://maurus.ttu.ee/idu0080/5/lahendus/bprocess_log_example.txt

Teenuste rakenduse logifail:

Iga teenus (3-st teenusest) peaks kirjutama informatsiooni sellesse logifaili oma tegevuse kohta.

1. Kullerifirmade teenus kirjutab faili järgmised andmed:

- käivitamise algus
- saadetud kullerfirmade nimekiri
- töö lõpu aja

2011.04.24 33:47 Courierservice starts
Courier: [DPD Sweden]
Courier: [Kiired]
Courier: [Kiirkuller]
Courier: [Tallinna Raskeveod]
Courier: [Tallinna Raskeveod]
Courier: [UPS Norway]
2011.04.24 33:47 Courierservice ends

2. Pakkumiste teenus kirjutab ühe pakkumise tegemisel:

- käivitamise algus
- saadetud kullerfirma id
- saadetud tellimuse hind
- ko haletoimetamise aadressi maakond
- kullerfirma protsent („percent_from_order” mis saadakse tund.ttu.ee

CourierService.getCourierById() väljakutsest)

- kui ko haletoimetamise aadress langeb kokku mõne kullerfirma aadressiga
- kui vähemalt üks müüja aadressidest langeb kokku ühe kullerfirma aadressiga
- pakkumisele tehtud allahindluse protsent (on suurem 0-st siis kui ko haletoimetamise aadress või mõni müüja aadressidest langeb kokku kullerfirma mõne aadressiga, seega kas 30% või 60%)
- transpordipakkumise id (genereeritakse pakkumise teenuse poolt)
- transpordipakkumise hind
- ko haletoimetamise aeg (random 1...7)
- töö lõpu aeg

2011.04.24 33:47 OfferService starts
received courier id: [4] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[20]
Shipping address langeb kokku kullerfirma aadressiga
Offer discount % [30%]
offer: offer_id: [D38AD66O7M] offer_price: [107.94] approx. delivery date: [Fri Apr 29 15:33:47 EEST 2011]
2011.04.24 33:47 OfferService ends

3. Transporditellimuse teenus kirjutab logifaili:

- käivitamise algus
- saadud pakkumise id (offer id)
- vastuseks saadetud saadetise „tracking number”
- töö lõpu aeg

2011.04.24 33:48 TransportOrderService starts
Pakkumisele (offer_id): [RA4RCKD61K] vastav tellimus registreeritud. Saadetise tracking number: [ra4rckd61k#20110424]
2011.04.24 33:48 TransportOrderService ends

Äriprotsessi rakenduse logifail:

Äriprotsess kirjutab logifaili:

- käivitamise algus
- tellimuse id (order id) ja hind
- pakkumiste teenuselt saadud pakkumiste koguarv
- pakkumiste andmed: pakkumise headus („rank”), pakkumise tegija (kullerfirma) nimi, transporditellimuse pakkumise hind, kohaletoimetamise päevade arv (mitme päeva pärast tuuakse), kohaletoimetamise kuupäev
- väljavalitud pakkumise id ja pakkumise teinud kullerfirma nimi , pakkumise id ja hind
- tellitud transporditeenuse tulemusena saadud saadetise „tracking number”
- tund.ttu.ee OrderShipment teenuse registreerimise tulemusena vastuseks saadud andmebaasi kirje nimber (mis on tund.ttu.ee serveris tabeli [order_shipment] kirje number)

2011.04.24 19:31 protsessi algus

order_id: 10 order price: 771.0

received 6 offers

rank: [1.5419999999999999] Courier: [DPD Sweden] Transport offer price: [154.2] days: [1]
approximate delivery date:[2011 Apr 25]

rank: [1.94292] Courier: [Kiirkuller] Transport offer price: [64.764] days: [3] approximate delivery
date:[2011 Apr 27]

rank: [2.4672] Courier: [Tallinna Raskeveod] Transport offer price: [123.36] days: [2] approximate
delivery date:[2011 Apr 26]

rank: [2.6214] Courier: [Tallinna Raskeveod] Transport offer price: [131.07] days: [2] approximate
delivery date:[2011 Apr 26]

rank: [5.7825] Courier: [UPS Norway] Transport offer price: [192.75] days: [3] approximate delivery
date:[2011 Apr 27]

rank: [6.1679999999999999] Courier: [Kiired] Transport offer price: [154.2] days: [4] approximate
delivery date:[2011 Apr 28]

Selected offer: courier: [DPD Sweden] offer id: [O4C6R57D6L] offer price: [154.2]

Tracking number: o4c6r57d6l#20110424

OrderShipment id: 36

2011.04.24 19:35 protsessi lopp

4.2. Pange rakendus tööle ka väljaspool Eclipse. Eksportige teneuste ja äriprotsessi projektid Eclipsest war-failidena ja installeerige need war failid oma arvutisse installeeritud Tomcat-i serverile (või kui on vajadus/võimalus siis arvutiklass arvutis käivitatud Tomcat-i alla). Kuidas see installeerimine aj käivitamine võiks välja näha sellest on juttu punktis 5.

5. Kuidas võiks välja näha töötava rakendus koduarvutis ja selle töötava rakenduse käivitamine ja töö

Oletame et meil on järgmised failid:

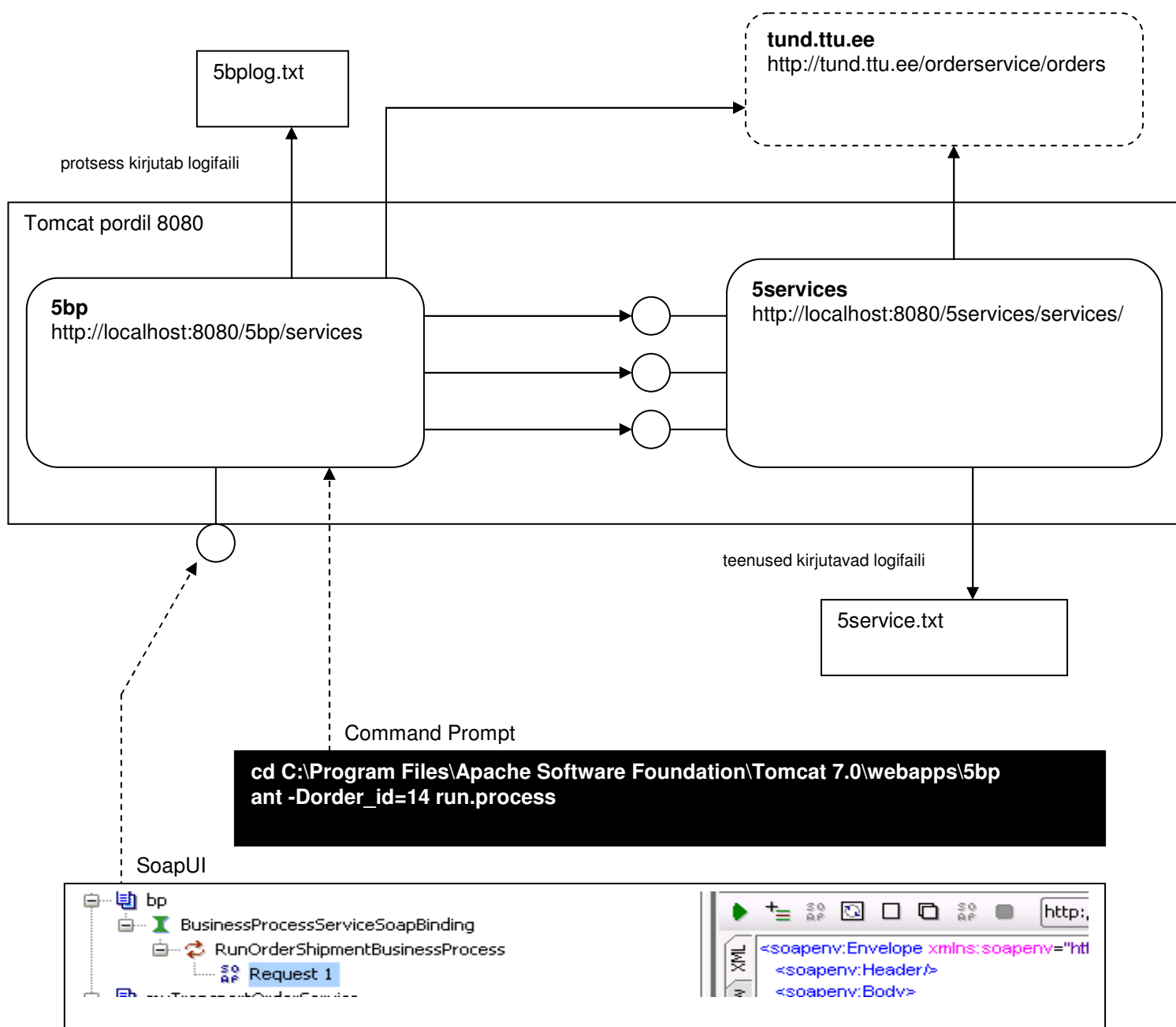
5services.war – lahenduste teenuste osa (kullerifirmade teenus, pakkumiste teenus, transporditeenuse tellimise teenus).

5bp – äriprotsess veebteenusena ja käsurealt (või Eclipsest) käivitavana.

build.xml – ant-i build fail äriprotsessi käivitamiseks käsurealt kui see build.xml kopeerida töötava veebirakenduse „5bp” kataloogi Tomcat-i serveril.

services_log_example.txt – teenuste rakenduse logifaili näide

bprocess_log_example.txt – äriprotsessi rakenduste logifaili näide



5.1. Lahenduse seadistamine ja käivitamine

Eeldame et lahendus on installeeritav Apache Tomcat-i keskkonda Windowsi ja Linux-i operatsioonisüsteemides.

Eeldused:

1. arvutisse on installeeritud Apache Tomcat.

Kuna rakendused ei kasuta andmebaasi siis pole vaja et arvutis oleks andmebaasisüsteem või et Tomcat-i /lib-kataloogis oleksid JDBC-draiverid.

2. Tomcat on installeeritud vaikimisi pordile 8080

Eeldame et äriprotsessi rakendused „5bp” on koodi sisse kirjutatud (õigem oleks küll öelda Eclipse poolt genereeritud viited) teenuste rakendusele aadressil <http://localhost:8080/5services/>

3. On olemas interneti-ühendus

Rakendus „5services” peab ühendama serveriga tund.ttu.ee.

4. Äriprotsessi käivitamiseks käsurealt on vajalik et arvutisse oleks installeeritud Apache ant (seega peaksid Windowsi all olema seadistatud keskkonnamuutujad **JAVA_HOME** ja Path peab lisaks viitama ant-i /bin-kataloogile)

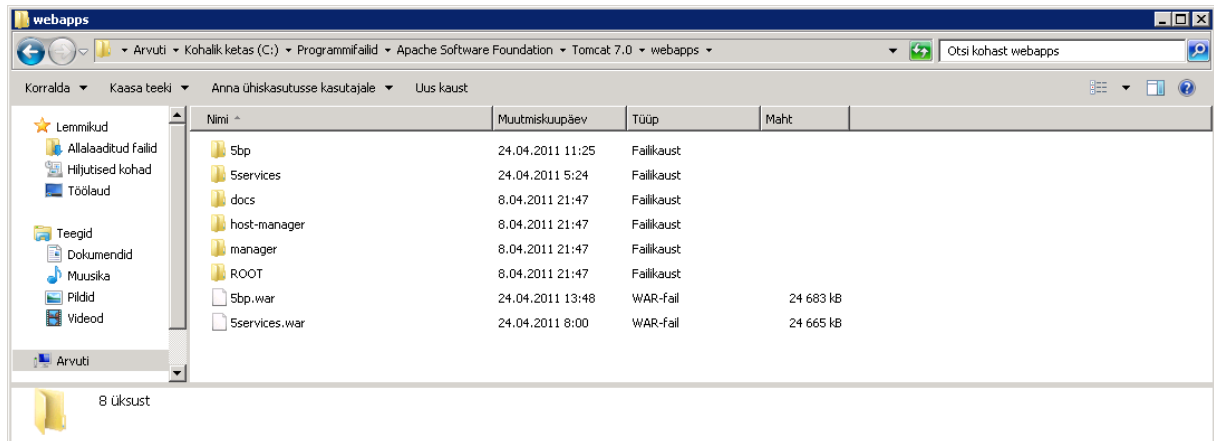
Nii et kokkuvõtteks : sobib Tomcat-i default installatsioon. Teil on Eclipsest tehtud teenuste ja äriprotsessi rakendused. Võite neid kaitsmisel käivitada ja jooksutada ka Eclipse seest aga võite proovida ka kas nad war-failidena Tomcat serveril käima lähevad (selleks tuleb Eclipse projektid eksportida war-failidena).

Alltoodud näide räägibki kuidas need teie kaks rakenduse osa võiksid Tomcat all (siis väljaspool Eclipse) tööle minna.

Installeerimine:

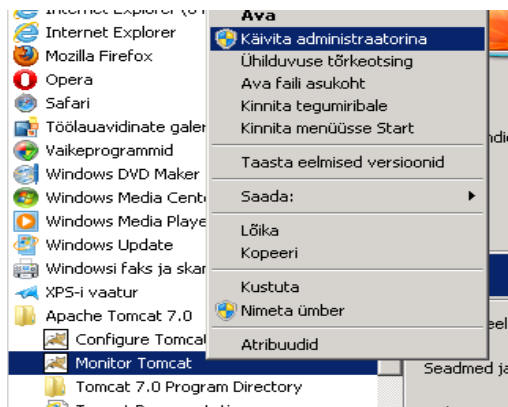
1. Kopeerige

rakenduste war-failid 5bp.war ja 5service.war oma Tomcati webapps kataloogi.

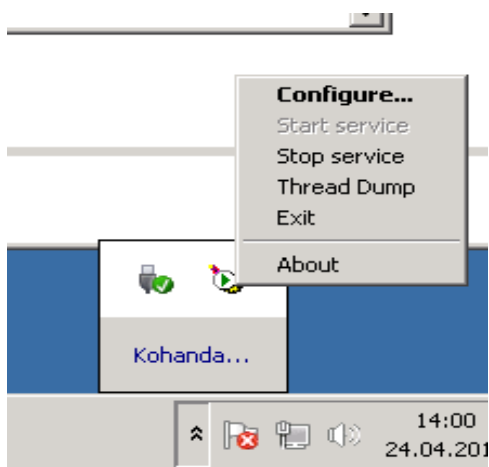


2. Käivitage Tomcat

Selleks valige Start menüüst „Monitor Tomcat” ja käivitage see „Rund as Administrator”-ina.



Nüüd saate vajadusel Tomcat-i seisma panna ja käivitada:



3. Vaadake kas teenused on üleval.

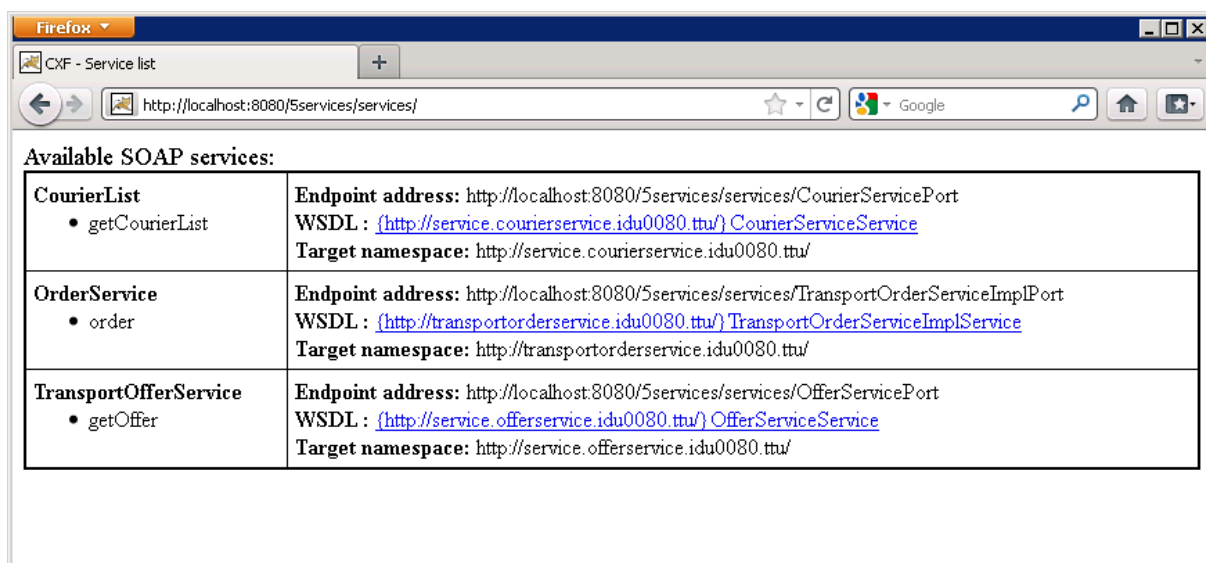
Oodake umbes pool minutit peale Tomcat-i käivitamist ja proovige kas saate midagi ette aadressidel

<http://localhost:8080/5services/services/>

ja

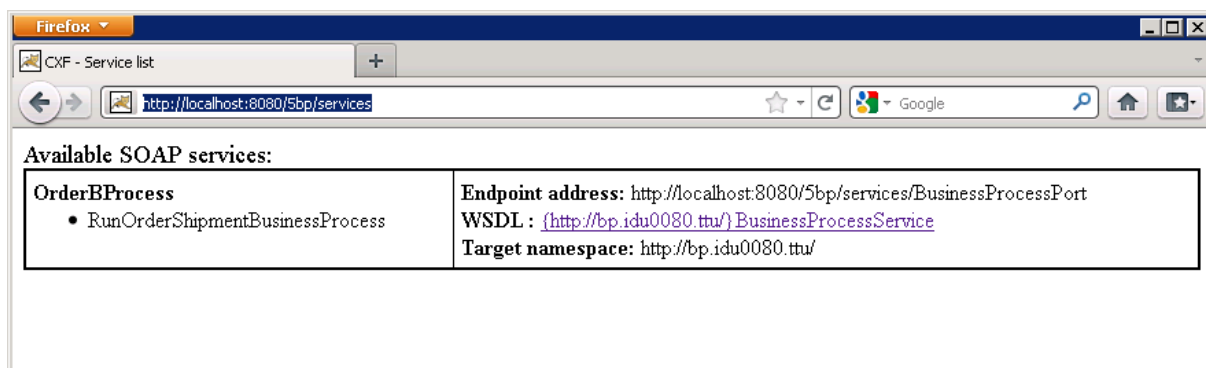
<http://localhost:8080/5bp/services/>

Pilt peaks olema umbes selline:



The screenshot shows a Firefox browser window with the address bar containing <http://localhost:8080/5services/services/>. The page content is titled "Available SOAP services:" and contains a table with three rows of service information.

Service Name	Endpoint address	WSDL	Target namespace
CourierList <ul style="list-style-type: none">getCourierList	http://localhost:8080/5services/services/CourierServicePort	http://service.courierservice.idu0080.ttu/ CourierServiceService	http://service.courierservice.idu0080.ttu/
OrderService <ul style="list-style-type: none">order	http://localhost:8080/5services/services/TransportOrderServiceImplPort	http://transportorderservice.idu0080.ttu/ TransportOrderServiceImplService	http://transportorderservice.idu0080.ttu/
TransportOfferService <ul style="list-style-type: none">getOffer	http://localhost:8080/5services/services/OfferServicePort	http://service.offerservice.idu0080.ttu/ OfferServiceService	http://service.offerservice.idu0080.ttu/



The screenshot shows a Firefox browser window with the address bar containing <http://localhost:8080/5bp/services/>. The page content is titled "Available SOAP services:" and contains a table with one row of service information.

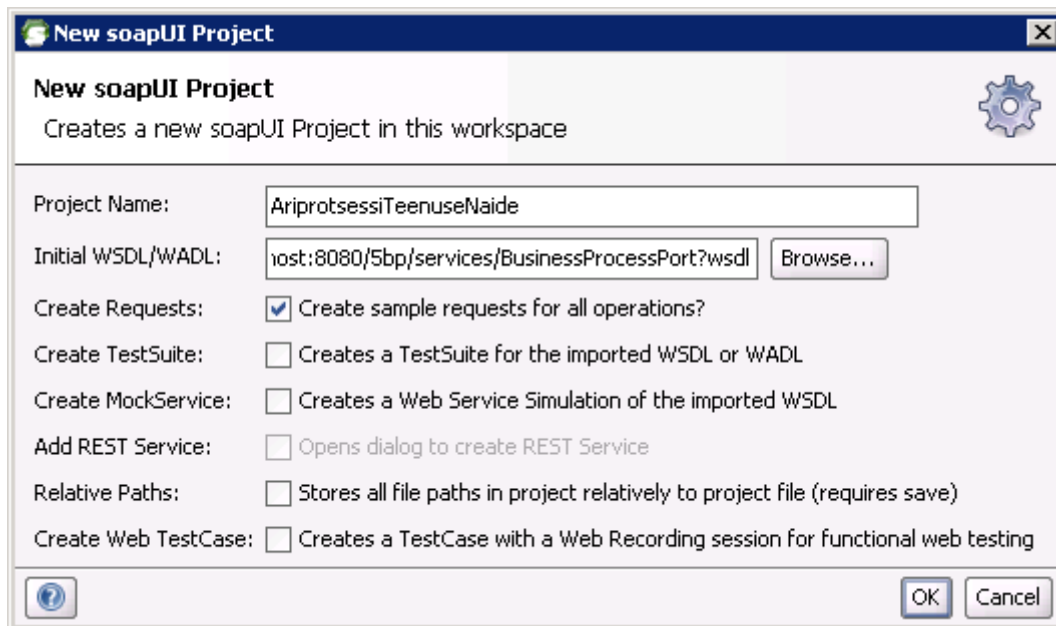
Service Name	Endpoint address	WSDL	Target namespace
OrderBProcess <ul style="list-style-type: none">RunOrderShipmentBusinessProcess	http://localhost:8080/5bp/services/BusinessProcessPort	http://bp.idu0080.ttu/ BusinessProcessService	http://bp.idu0080.ttu/

4. Testimine SoapUI-ga.

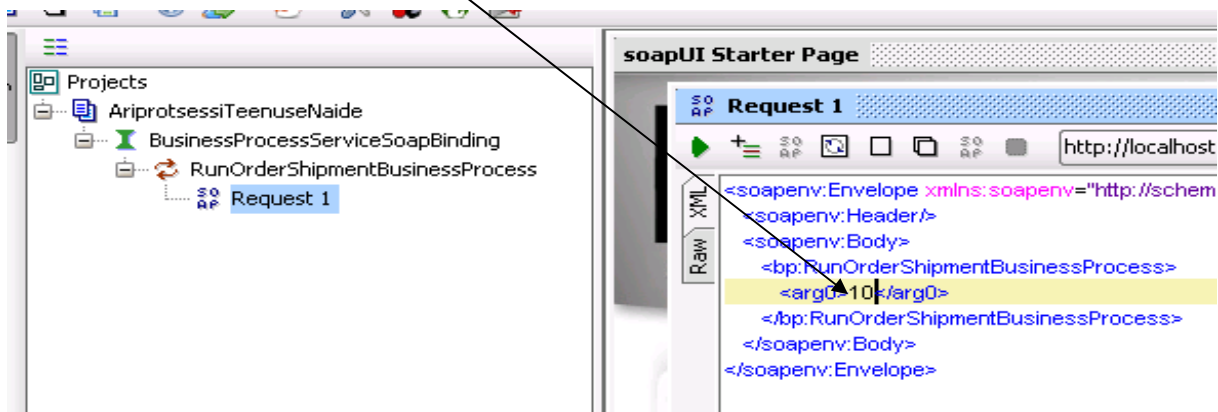
Kopeerige lehelt aadressil <http://localhost:8080/5bp/services/> äriprotsessi teenuse wsdl-i link ja käivitage SoapUI.

<http://localhost:8080/5bp/services/BusinessProcessPort?wsdl>

Tehke uus SoapUI projekt ja sisestage teenuse wsdl-i aadress.



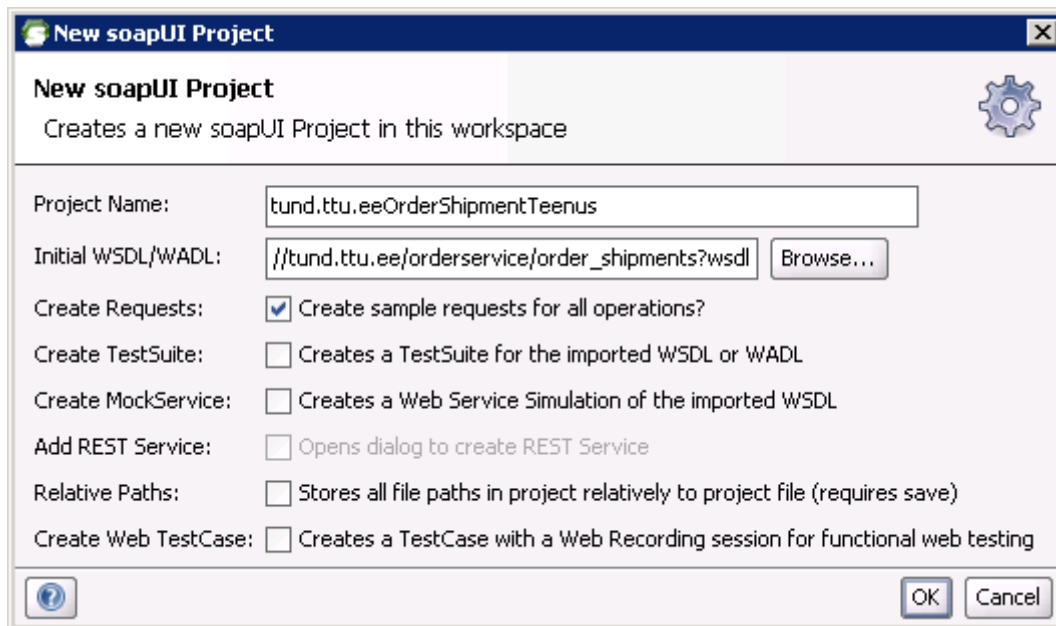
Käivitage äriprotsessi teenus andes argumendiks tellimuse id (order_id) millele soovitakse transporditeenust valida ja tellida.



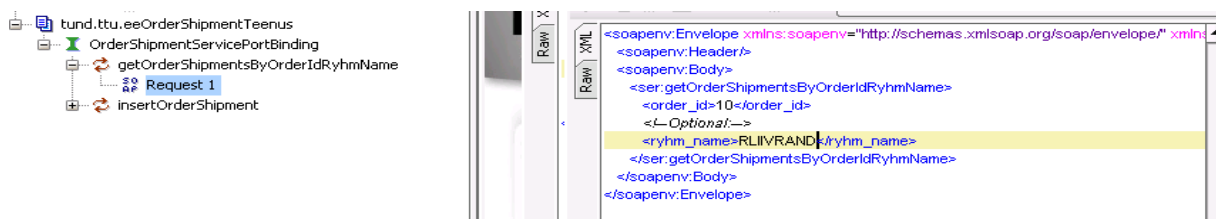
Nüüd tehke teine SoapUI projekt tulemuste vaatamiseks tund.ttu.ee veebteenusest.

Teenuse wsdl-i address on

http://tund.ttu.ee/orderservice/order_shipments?wsdl



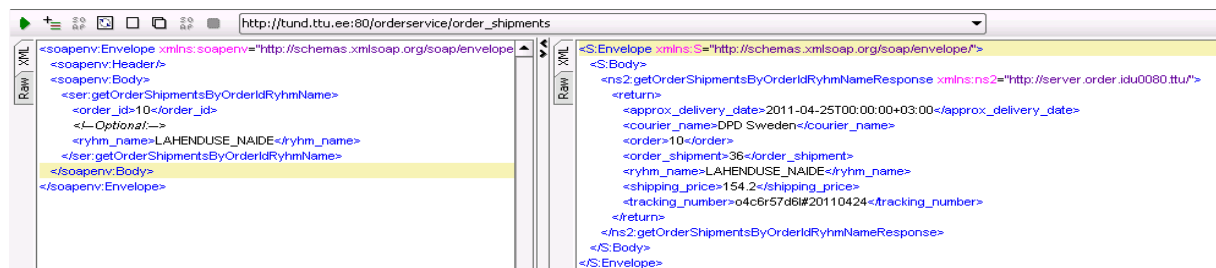
Sellest teenusest on meil praegu vaja meetodit **getOrderShipmentsByOrderIdRyhmiName()**



Sisestame parameetri order_id väärtuseks selle tellimuse id mille oma äriprotsessi sisendisse andsime (ehk order_id=10 praeguses näites).

Ryhmi_name väärtuseks paneme LAHENDUSE_NAIDE (see on antud näite lähtekoodi sisse kirjutatud – tellimustele transpordi registreerimisel tund.ttu.ee SOAP-teenuse abil on ryhmi_name="LAHENDUSE_NAIDE")

```
private String ryhmiName="LAHENDUSE_NAIDE";
```



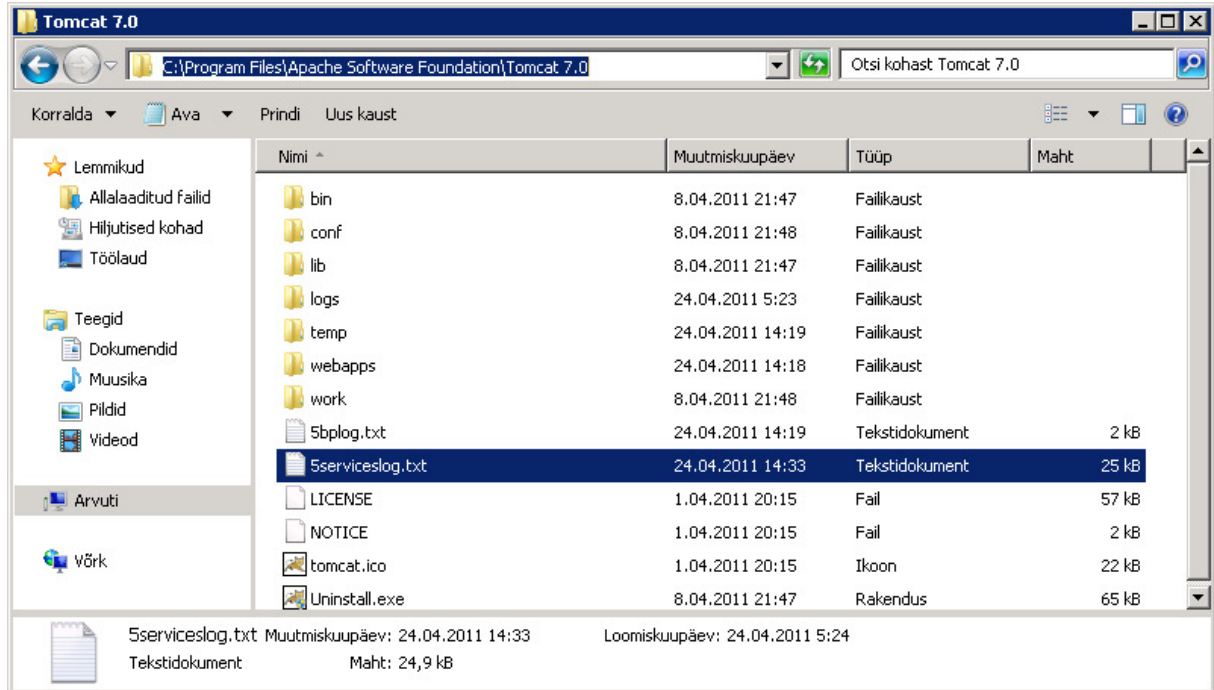
Teie lahenduses on rühma nimeks teie rühma nimi mis võib ka olla rakendusse konstandina sisse kirjutatud.

SOAP-pöördumise tulemusena saate tund.ttu.ee PostgreSQL-i andmebaasi registreeritud tellimusega seotud transpordi-teenuse andmed mis PostgreSQL-i andmebaasi kirjena näevad välja nii:

	order_shipment [PK] numeric(10,0)	order_fk numeric(10,0)	tracking_number text	courier_name text	shipping_price numeric	approx_delivery_date date	created timestamp without ti	ryhm_name text
1	36	10	o4c6r57d61#20110424	DPD Sweden	154.2	2011-04-25	2011-04-24 17:24	LAHENDUSE_NAIDE
*								

Teenuste rakenduse ja äriprotsessi rakenduse logifailid:

Teenuste rakenduse logifaili nimeks on „5serviceslog.txt” ja see tekib Tomcat-i juurkataloogi:



5serviceslog.txt

2011.04.24 33:47 Courierservice starts
Courier: [DPD Sweden]
Courier: [Kiired]
Courier: [Kiirkuller]
Courier: [Tallinna Raskeveod]
Courier: [Tallinna Raskeveod]
Courier: [UPS Norway]
2011.04.24 33:47 Courierservice ends

2011.04.24 33:47 OfferService starts
received courier id: [4] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[20]
Shipping address langeb kokku kullerfirma aadressiga
Offer discount % [30%]
offer: offer_id: [D38AD66O7M] offer_price: [107.94] approx. delivery date: [Fri Apr 29 15:33:47 EEST 2011]
2011.04.24 33:47 OfferService ends

2011.04.24 33:47 OfferService starts
received courier id: [6] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[20]
Offer discount % [0%]
offer: offer_id: [SG4X84KM8R] offer_price: [154.2] approx. delivery date: [Mon Apr 25 15:33:47 EEST 2011]
2011.04.24 33:47 OfferService ends

2011.04.24 33:47 OfferService starts
received courier id: [3] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[12]
Myyja address langeb kokku kullerfirma aadressiga

Offer discount % [30%]
offer: offer_id: [RA4RCKD61K] offer_price: [64.764] approx. delivery date: [Tue Apr 26 15:33:47 EEST 2011]
2011.04.24 33:47 OfferService ends

2011.04.24 33:47 OfferService starts
received courier id: [7] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[17]
Offer discount % [0%]
offer: offer_id: [RPML8LXGKC] offer_price: [131.07] approx. delivery date: [Tue Apr 26 15:33:47 EEST 2011]
2011.04.24 33:47 OfferService ends

2011.04.24 33:48 OfferService starts
received courier id: [8] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[16]
Offer discount % [0%]
offer: offer_id: [5A8G3SKLOK] offer_price: [123.36] approx. delivery date: [Thu Apr 28 15:33:48 EEST 2011]
2011.04.24 33:48 OfferService ends

2011.04.24 33:48 OfferService starts
received courier id: [5] order price: [771.0] shipping address county: [Norbotten]
Kullerfirma protsent:[25]
Offer discount % [0%]
offer: offer_id: [5ZAPXZORD4] offer_price: [192.75] approx. delivery date: [Fri Apr 29 15:33:48 EEST 2011]
2011.04.24 33:48 OfferService ends

2011.04.24 33:48 TransportOrderService starts
Pakkumisele (offer_id): [RA4RCKD61K] vastav tellimus registreeritud. Saadetise tracking number: [ra4rckd61k#20110424]
2011.04.24 33:48 TransportOrderService ends

Äriprotsessi rakenduse logifaili nimeks on „5bplog.txt” ja ka see tekib Tomcat-i juurkataloogi:

5bplog.txt

2011.04.24 19:31 protsessi algus

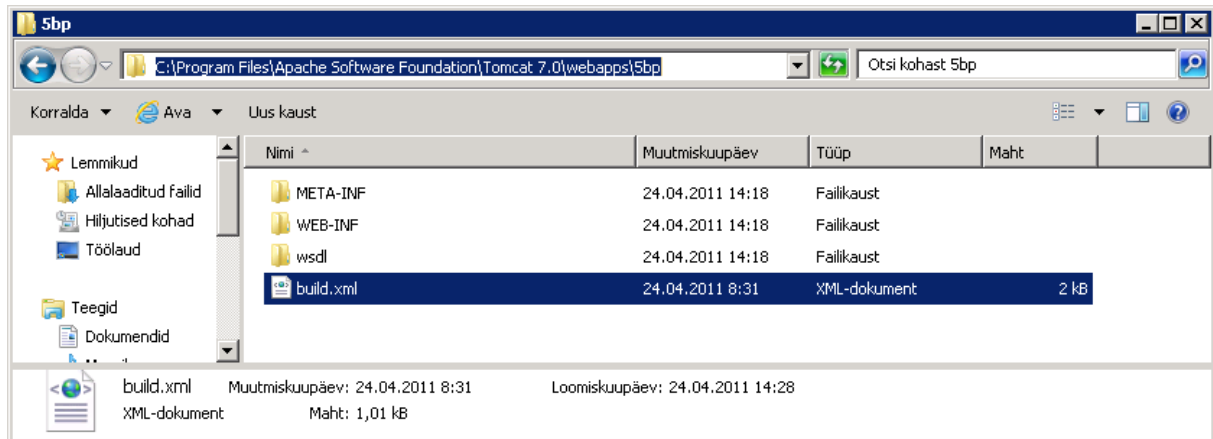
order_id: 10 order price: 771.0
received 6 offers

rank: [1.5419999999999998] Courier: [DPD Sweden] Transport offer price: [154.2] days: [1] approximate delivery date:[2011 Apr 25]
rank: [1.94292] Courier: [Kiirkuller] Transport offer price: [64.764] days: [3] approximate delivery date:[2011 Apr 27]
rank: [2.4672] Courier: [Tallinna Raskeveod] Transport offer price: [123.36] days: [2] approximate delivery date:[2011 Apr 26]
rank: [2.6214] Courier: [Tallinna Raskeveod] Transport offer price: [131.07] days: [2] approximate delivery date:[2011 Apr 26]
rank: [5.7825] Courier: [UPS Norway] Transport offer price: [192.75] days: [3] approximate delivery date:[2011 Apr 27]
rank: [6.167999999999999] Courier: [Kiired] Transport offer price: [154.2] days: [4] approximate delivery date:[2011 Apr 28]

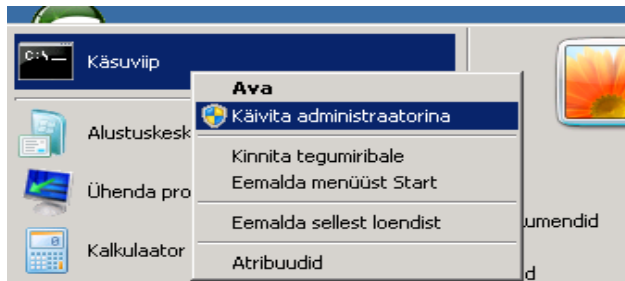
Selected offer: courier: [DPD Sweden] offer id: [O4C6R57D6L] offer price: [154.2]
Tracking number: o4c6r57d6l#20110424
OrderShipment id: 36
2011.04.24 19:35 protsessi lopp

Näite äriprotsessi käivitamine käsuraalt.

1. kopeerige kataloogist <http://maurus.ttu.ee/idu0080/5/lahendus/> võetud build.xml fail veebirakenduse „5bp” juurkataloogi Tomcat-i serveril.

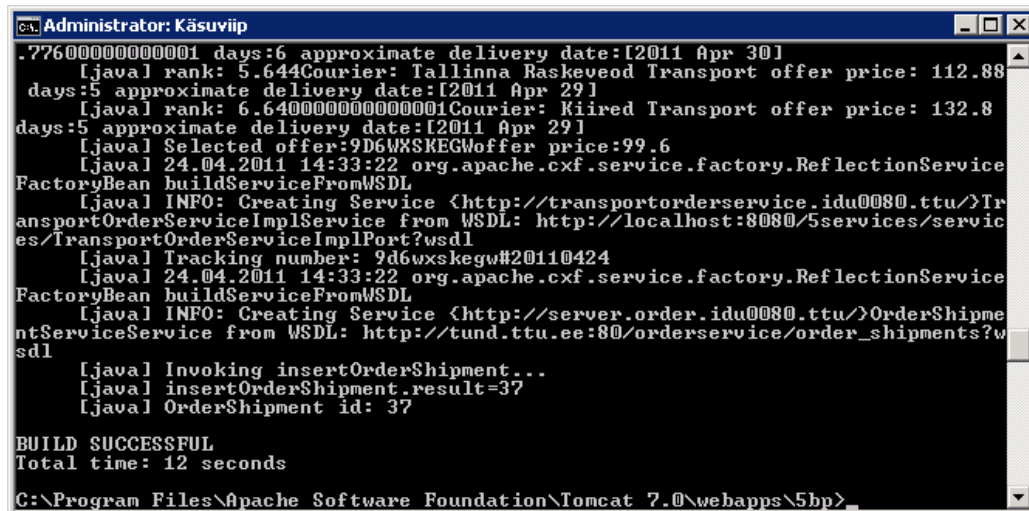


2. avage „Command Prompt” „Run as Administrator”-ina (mitte-administraatori režiimis ei saa rakendus logifaili kirjutada veebirakenduse kataloogi).



Käivitade järgmised käsud:

```
cd C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\5bp
ant -Dorder_id=14 run.process
```



Käivitamise tulemusi tund.ttu.ee serveris saab nüüd kontrollida jälle SoapUI-ga.

Äriprotsessi logifail tekib käsurealt käivitamisel nüüd teise kataloogi (mitte samasse kataloogi kuhu tekib äriprotsessi käivitamisel veebteenusena) . Logi tekib nüüd samasse kataloogi kus käivitamine toimus (mitte enam Tomcati juurkataloogi):

C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\5bp

