

# Solution to drop 2

## INF5150, Autumn 2005

Submitted by project group 1:

Robøle Torunn

Thorvaldsen Kjersti

Stoltenberg Vibeke

Koudrik Igor

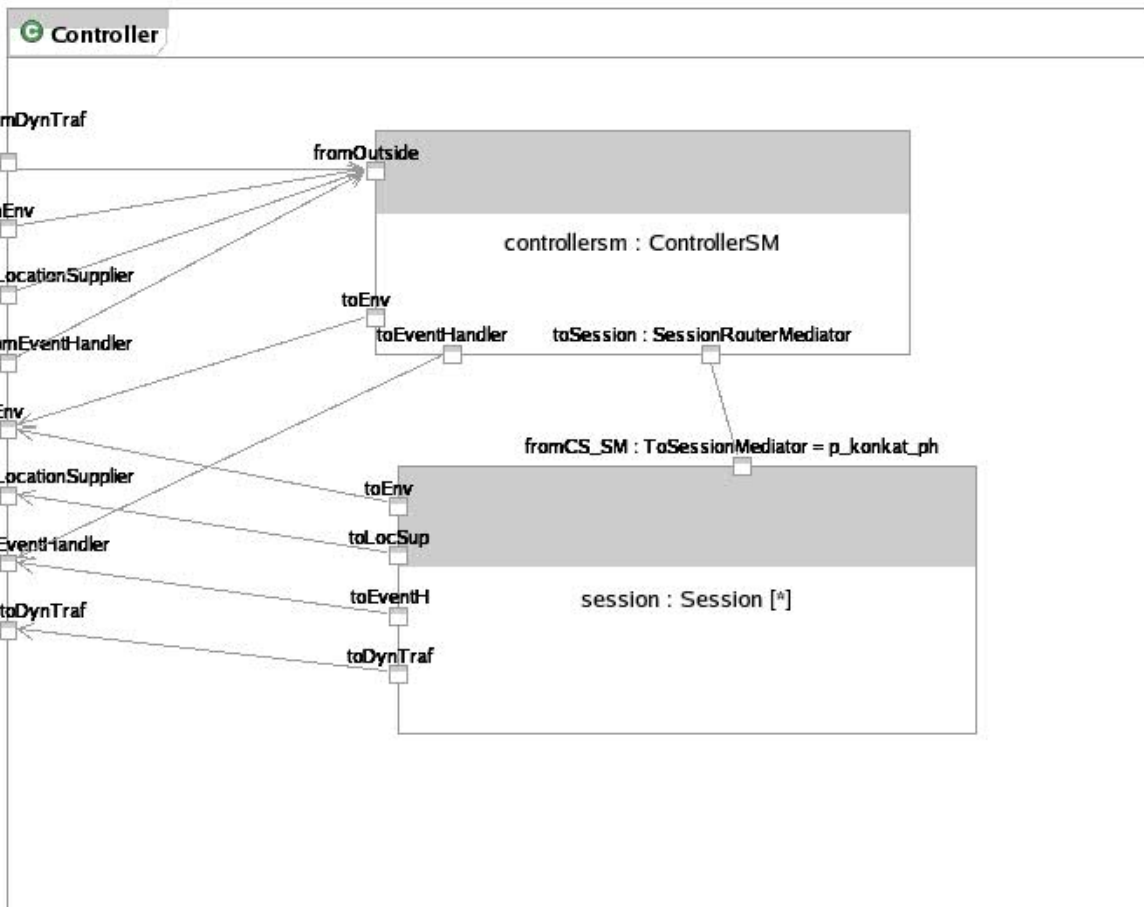
## **This document contains:**

1. Instructions to run the system
2. Model diagram report for executable UML design of Drop1
3. How we may understand the design as a refinement of the Drop 1 specification
4. Security analysis of Drop1 using CORAS tool

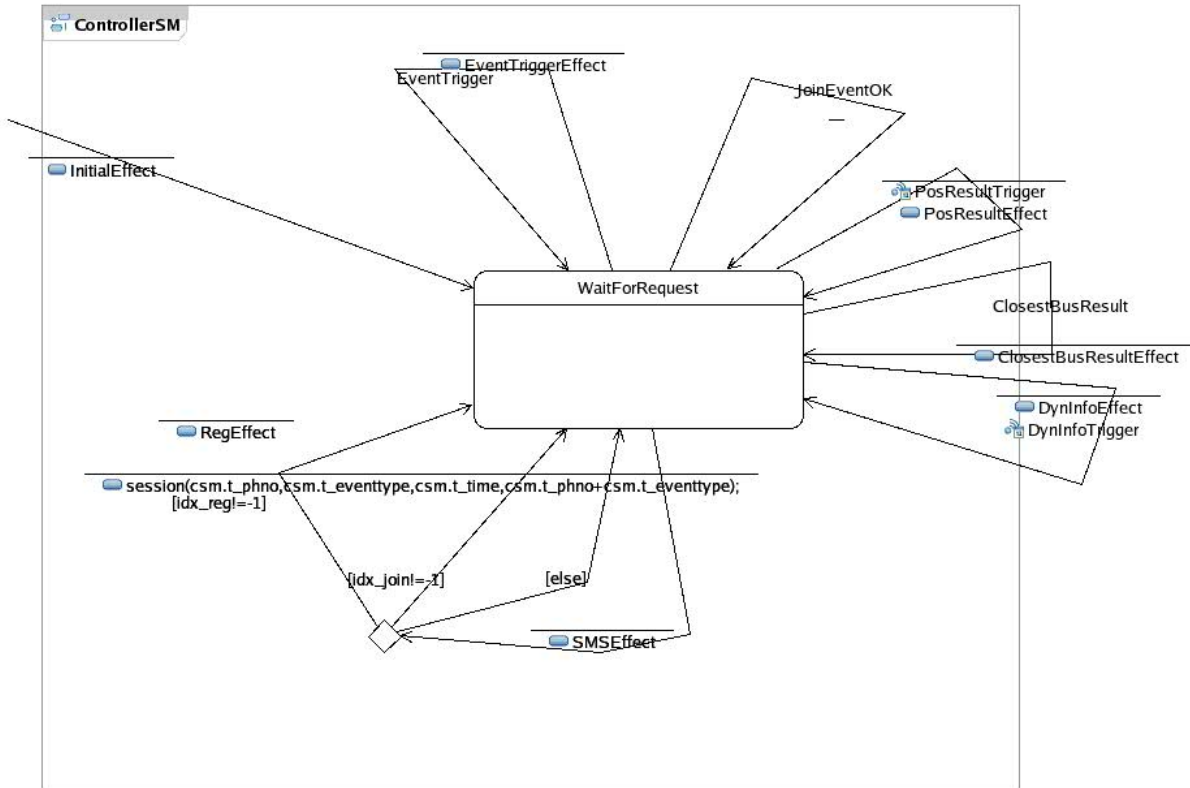
**Instructions to run the system**

1. The system is started by launching Drop2\_fat.jar file.
2. SMS message should be of the following format: stud1 konto user operation event time, where 'user' is a username at IFI, 'operation' is either 'register' or 'join', and 'event' is either 'bar' or 'kino', 'time' is time for the event.





BDPkg::BDPkg::ControllerSM::Diagram1 -- Statechart Diagram

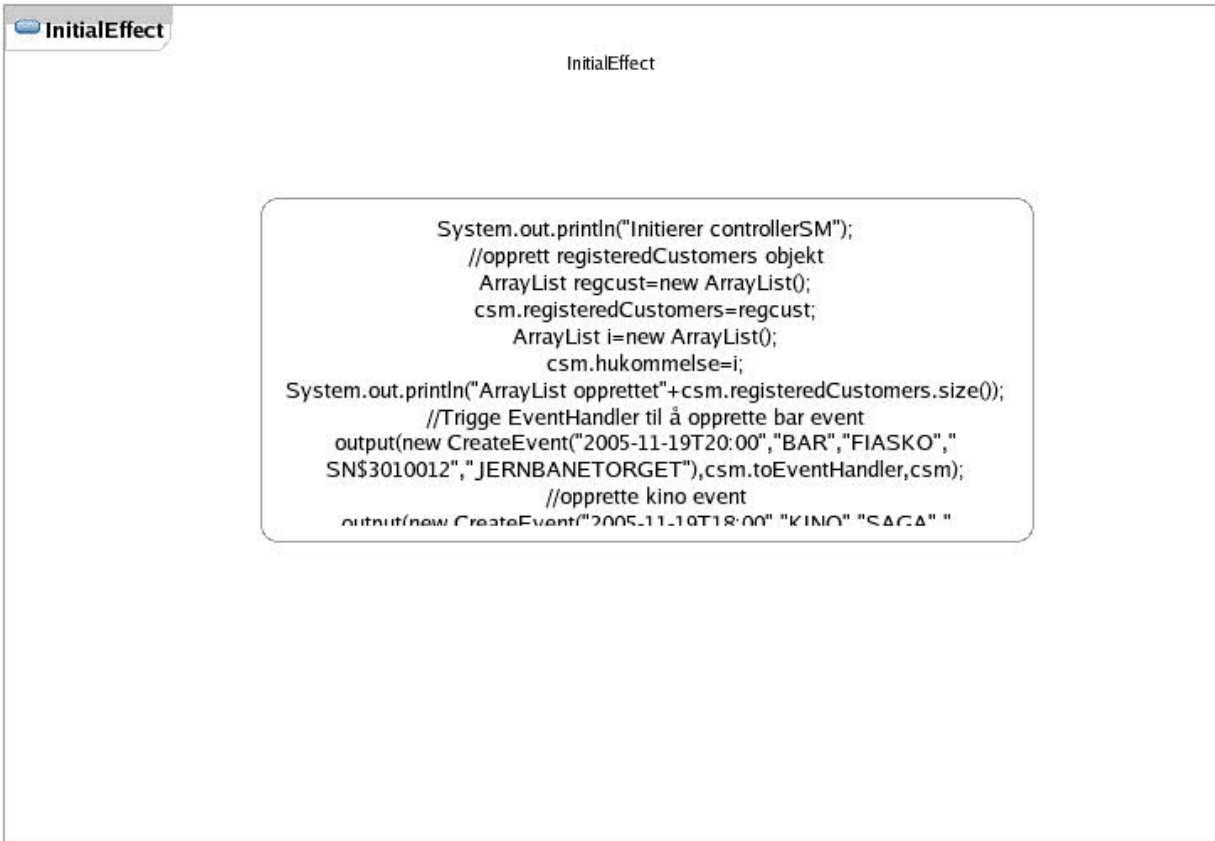


BDPkg::BDPkg::ControllerSM::Region1::<Transition>::DynInfoEffect::Diagram1 --  
Activity Diagram

#### DynInfoEffect

```
//hente ut phoneNo fra DynInfo, så kan route til rett session
System.out.println("RoutingInfo: "+sig.getRoutingInfo());
    String konkat=sig.getRoutingInfo();
    String ph=konkat.substring(0,8);
    //System.out.println("RoutingInfo="+ph);
    //hente ut timestamp
    String timestamp=sig.getTimeStamp();
    //sjekke om det skjedde feil ved henting
    //System.out.println("status="+sig.getStatus());
    String status=sig.getStatus();
    if(status.equals("ok"))
    {
        //søket har gått bra, kan sende videre ruteinfo
        //hente ut DynRoute
        DynRoute[] dynRoute=sig.getDynRoutes();
        System.out.println("Antall avqanqer="+dynRoute.length);
```

BDPkg::BDPkg::ControllerSM::Region1::<Transition>::InitialEffect::Diagram1 -- Activity Diagram



BDPkg::BDPkg::ControllerSM::Region1::<Transition>::PosResultEffect::Diagram1 --  
Activity Diagram



**PosResultEffect**

## PosResultEffect

```
//int ph=Integer.parseInt(sig.getMessageId());
String posRes=sig.getPositioningResult();
System.out.println("posResult="+sig.getPositioningResult());
//finne part "phone"
int idx_ph_1=posRes.indexOf("<STATICID>");
int idx_ph_2=posRes.indexOf("</STATICID>");
//System.out.println("idx1="+idx_ph_1+" idx2="+idx_ph_2);
String ph=posRes.substring(idx_ph_1+10,idx_ph_2);
//henter ut konkat fra hukommelse
boolean slutt=false;
String konkat="";
int s=csm.hukommelse.size();
for(int j=0;j<s && !slutt;j++)
{
String sub=(String)csm.hukommelse.get(j);
System.out.println("sub="+sub);
String ph_loop=sub.substring(0,8);
System.out.println("ph_loop"+ph_loop);
if (ph_loop.equals(ph))
{
System.out.println("i if for "+j);
konkat=sub;
}
}
}
```

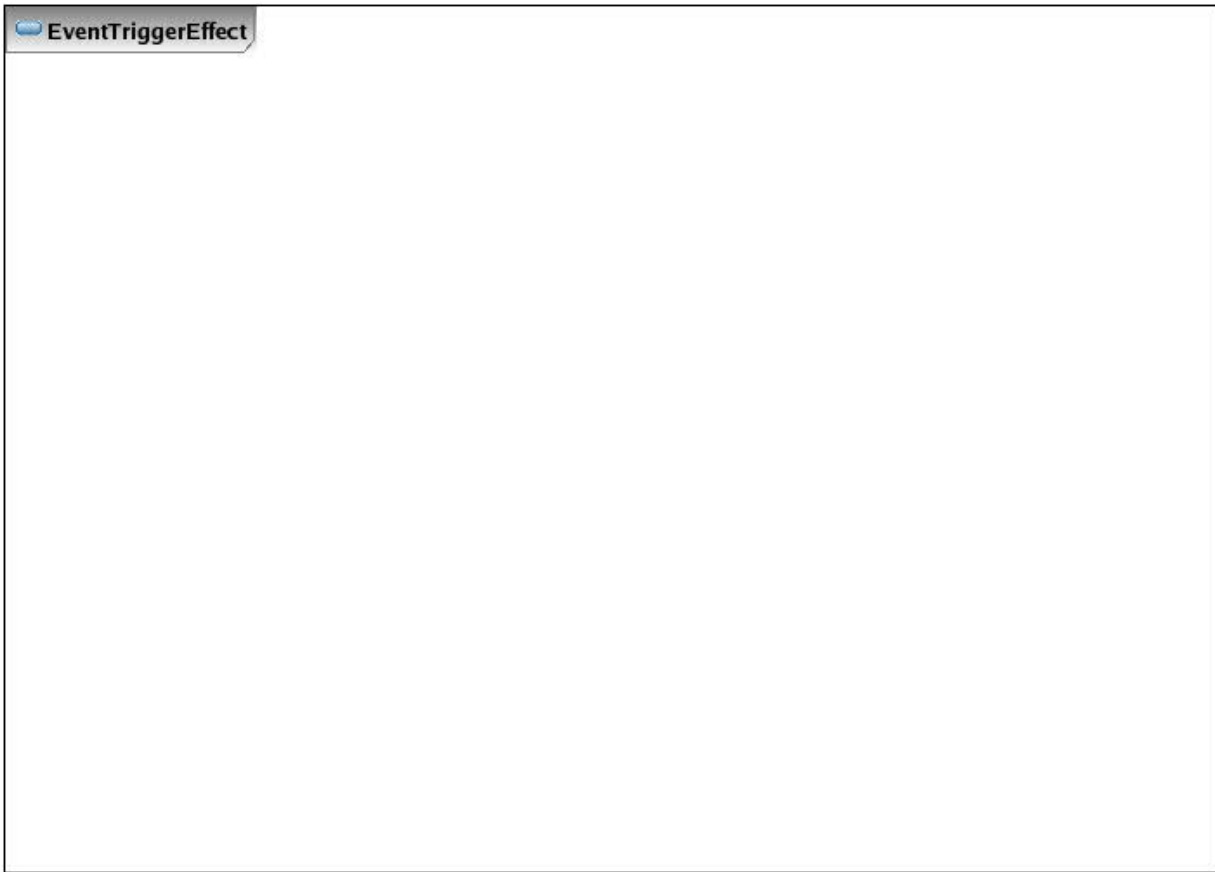
BDPkg::BDPkg::ControllerSM::Region1::<Transition>::SMSEffect::Diagram1 -- Activity Diagram

```
SMSEffect

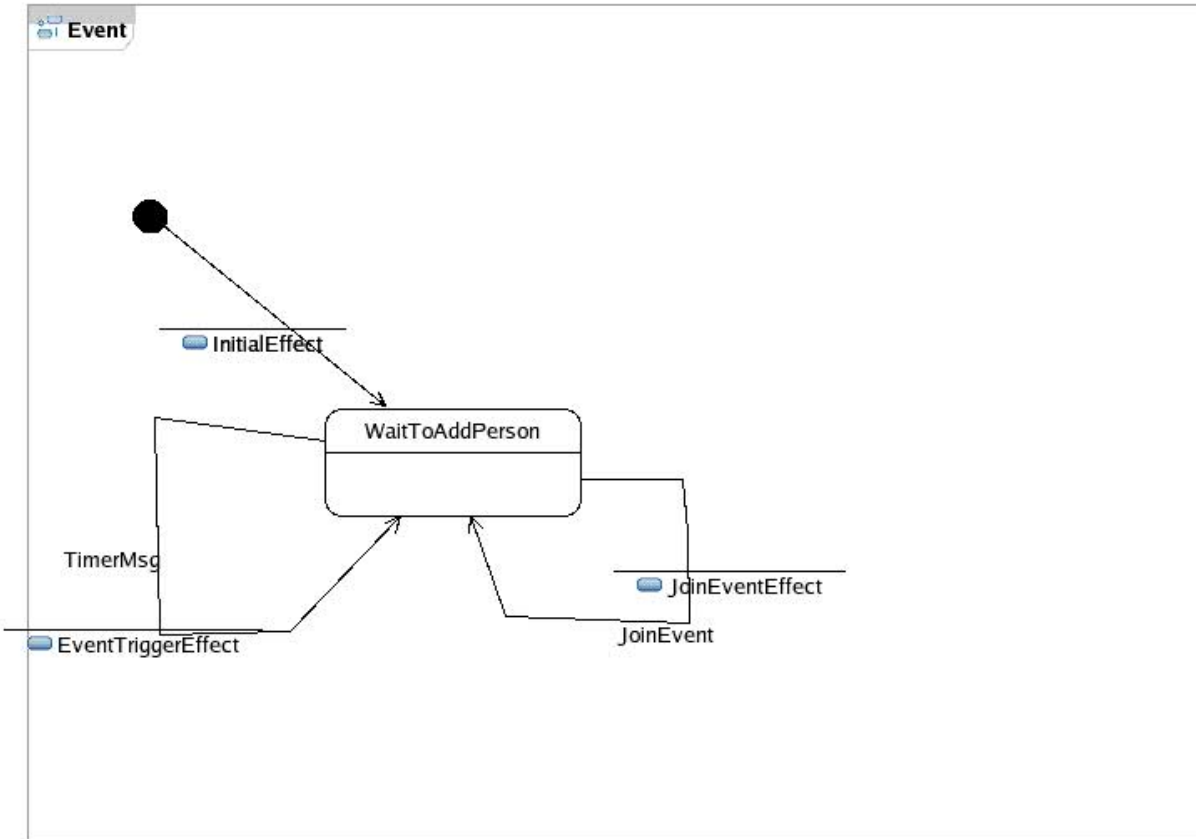
System.out.println("I SMSEffect. sig.msg="+sig.getMessage());System.out.println("I SMSEffect. sig.messageContent="+
    sig.messageContent());
//csm.t_phno=Integer.parseInt(sig.getFrom());
csm.t_phno=sig.getFrom();
System.out.println("from="+sig.getFrom());

//finne hvilken type mld det er
int idx_join=smstext.indexOf("JOIN");
//System.out.println("idx_join="+idx_join);
int idx_reg=smstext.indexOf("REGISTER");
//System.out.println("idx_reg="+idx_reg);
//hvis resultat fra indexOf er -1 betyr det at det ikke er treff
if (idx_join!=-1)
{
//det er join - henter ut verdi for eventtype
int idx_event=smstext.indexOf(" ",idx_join+5);
csm.t_eventtype=smstext.substring(idx_join+5,idx_event);
//hente ut verdi for time
int idx_time=smstext.indexOf(" ",idx_event);
csm.t_time=smstext.substring(idx_event+1);
System.out.println("t_eventType="+csm.t_eventtype+" t_time="+csm.t_time);
}
else if (idx_reg!=-1)
{
//det er reg - hente ut navn
System.out.println(smstext.substring(idx_reg+9));
csm.t_navn=smstext.substring(idx_reg+9);
//System.out.println("Før opprettet phone t_phno"+csm.t_phno);
PhoneNo ph=new PhoneNo();
ph=new PhoneNo(csm.t_phno);
Customer cust=new Customer(ph,csm.t_navn);
//System.out.println("size="+csm.registeredCustomers.size());
boolean conf=csm.registeredCustomers.add(cust);
System.out.println("t_navn="+csm.t_navn+" t_ph="+csm.t_phno);
}
}
```

BDPkg::BDPkg::ControllerSM::Region1::EventTrigger::EventTriggerEffect::Diagram1 -- Activity Diagram



BDPkg::BDPkg::Event::Diagram1 -- Statechart Diagram



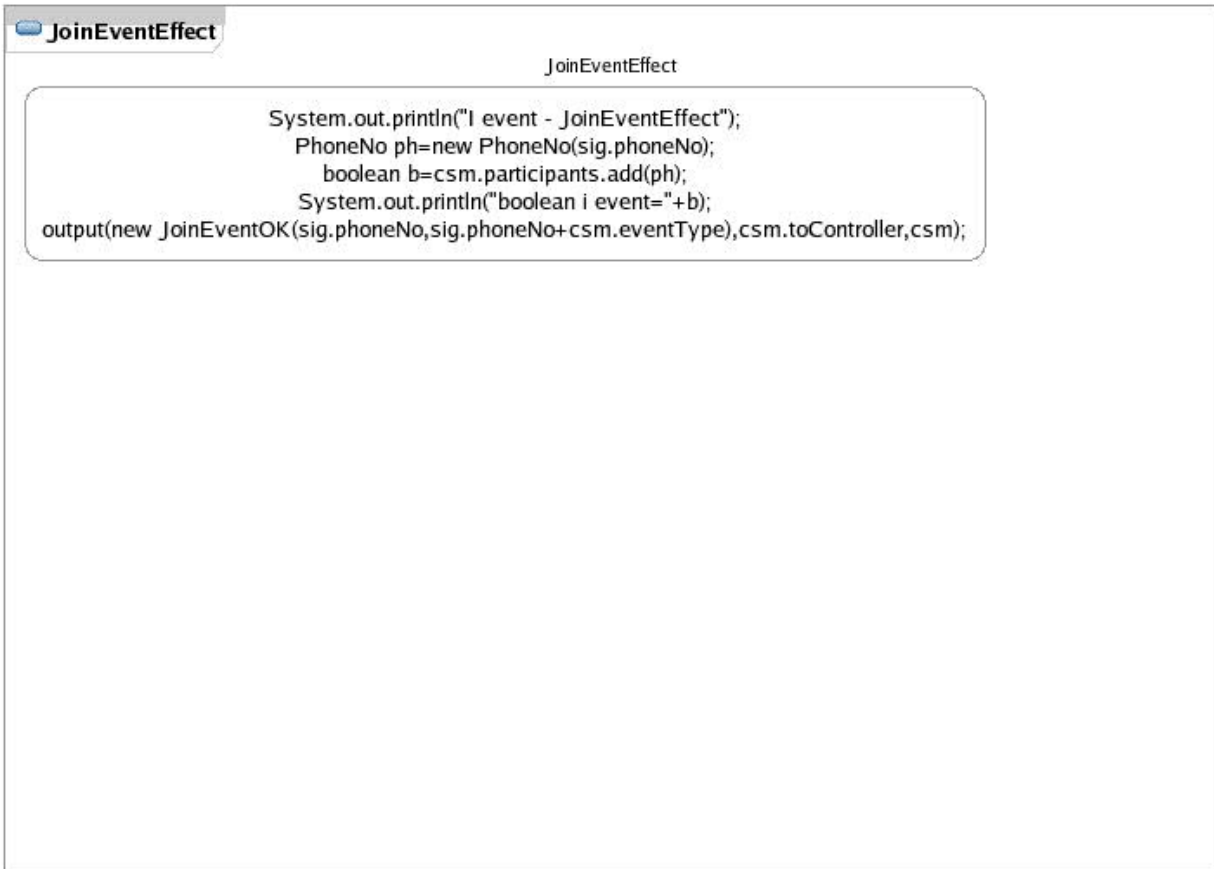
BDPkg::BDPkg::Event::Region1::<Transition>::InitialEffect::Diagram1 -- Activity Diagram

InitialEffect

InitialEffect

```
System.out.println("Event initieres");  
//starter timer til å trigge om 5 min fra event opprettet  
csm.timerMsg.startTimer(100000);  
    ArrayList part=new ArrayList();  
        csm.participants=part;  
csm.eventType=csm.p_eventType;  
csm.eventStartTime=csm.p_eventStartTime;  
csm.locName=csm.p_locName;  
csm.destBusStopID=csm.p_destBusStopID;  
csm.destBusStopName=csm.p_destBusStopName;  
System.out.println("eventType laget er: "+csm.eventType);
```

BDPkg::BDPkg::Event::Region1::JoinEvent::JoinEventEffect::Diagram1 -- Activity Diagram

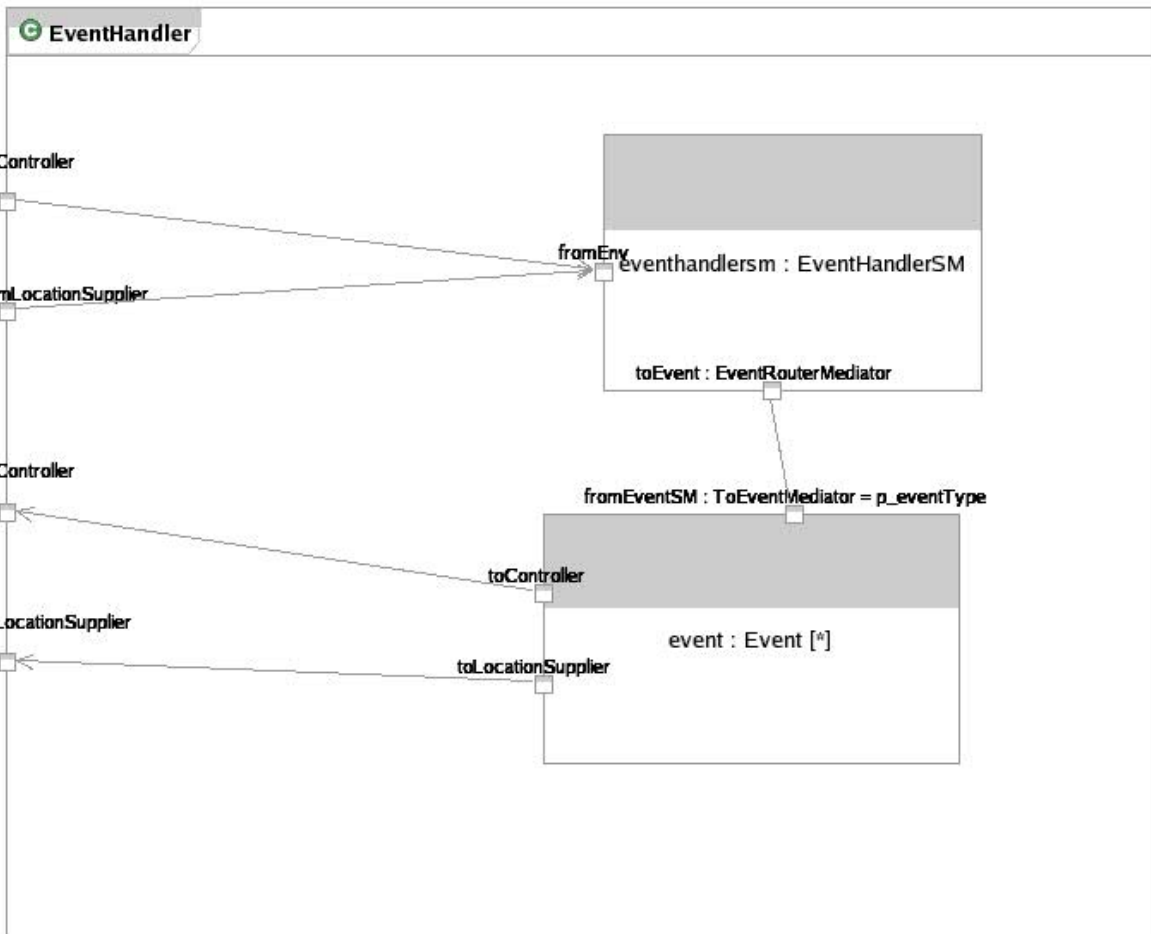


BDPkg::BDPkg::Event::Region1::TimerMsg::EventTriggerEffect::Diagram1 -- Activity Diagram

## EventTriggerEffect

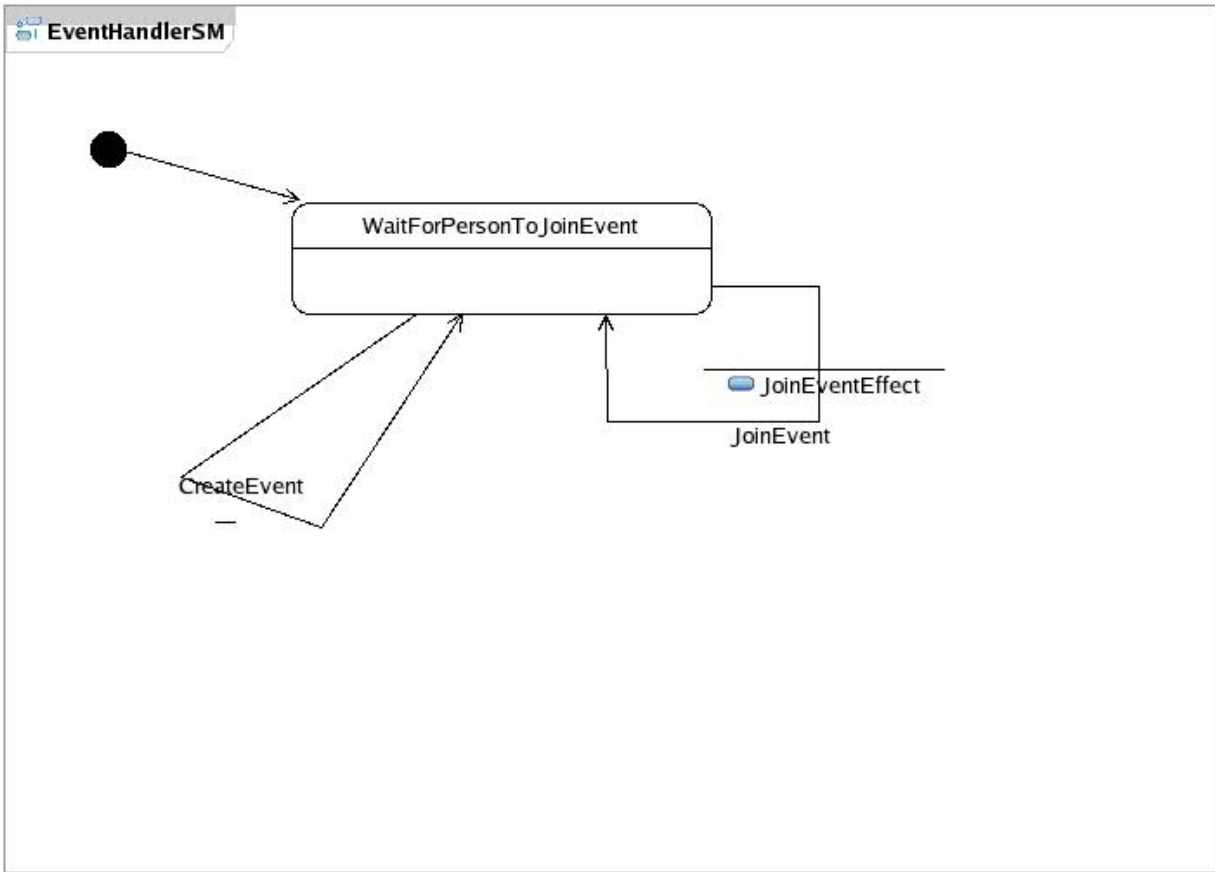
```
System.out.println("Event er trigget");
//henter alle deltakere som er påmeldt og sender trigger til deres session via controllersm
System.out.println("Antall participants: "+csm.participants.size());
for (int i=0;i<csm.participants.size();i++)
{
    PhoneNo ph=new PhoneNo();
    ph=(PhoneNo)csm.participants.get(i);
    System.out.println("Participant phone: "+ph.number);
    output(new EventTrigger(ph.number,csm.eventType,csm.locName,csm.destBusStopID,csm.
    destBusStopName,ph.number+csm.eventType), csm.toController,csm);
}
//fjerner data fra participants
csm.participants.clear();
```

BDPkg::BDPkg::EventHandler::Diagram1 -- Structure Diagram

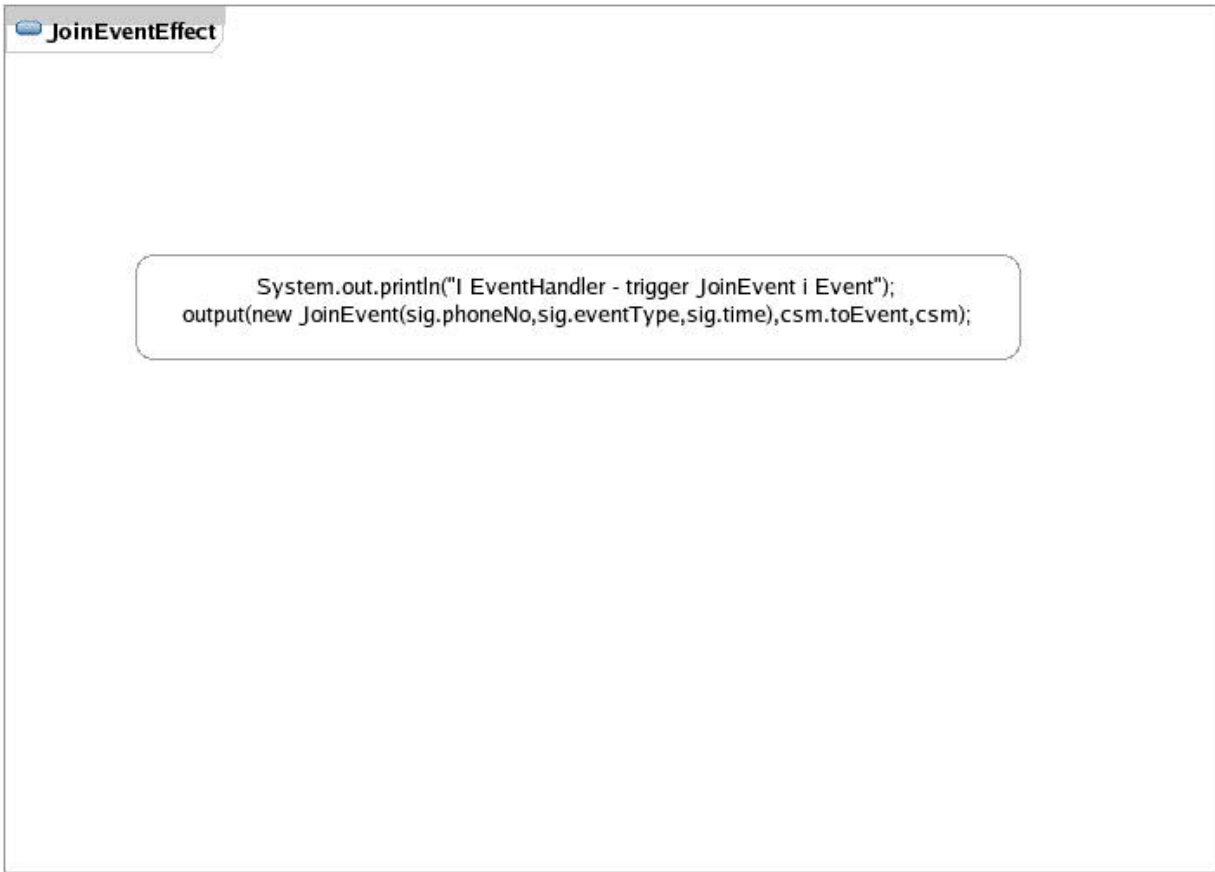


BDPkg::BDPkg::EventHandlerSM::Diagram1 -- Statechart Diagram

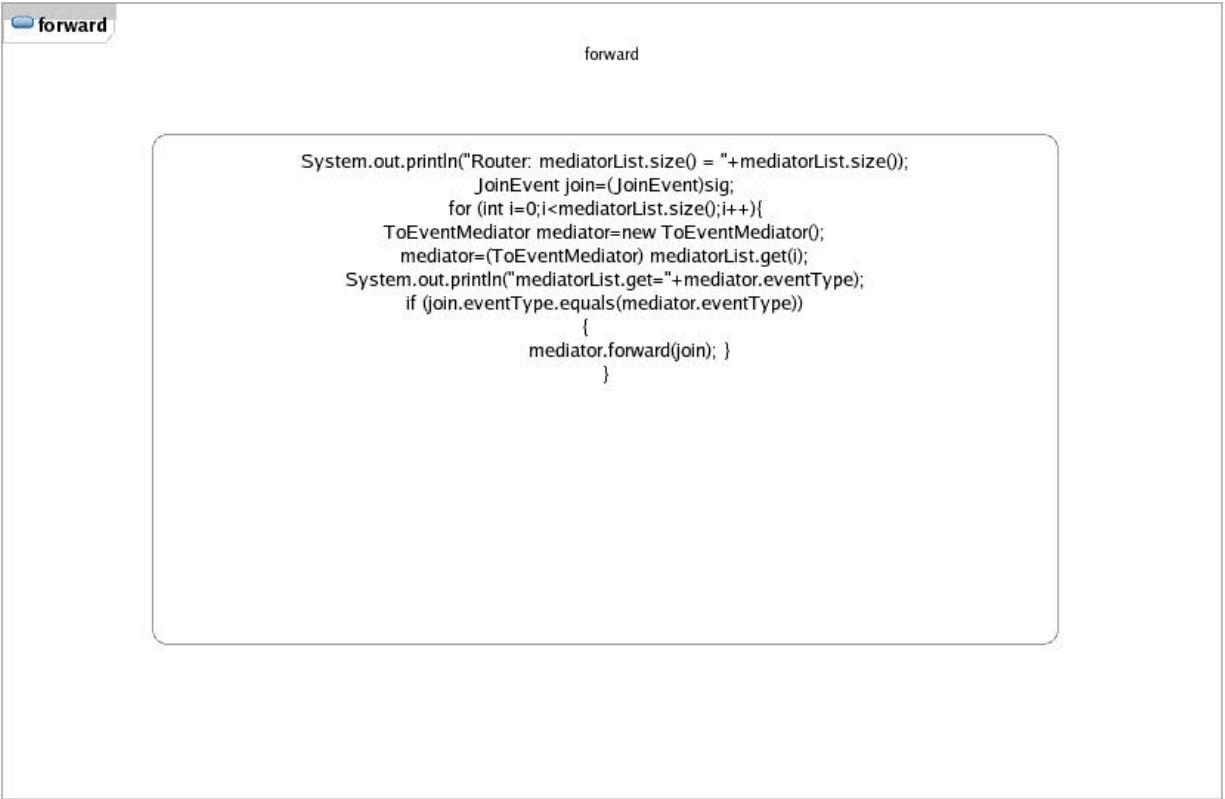




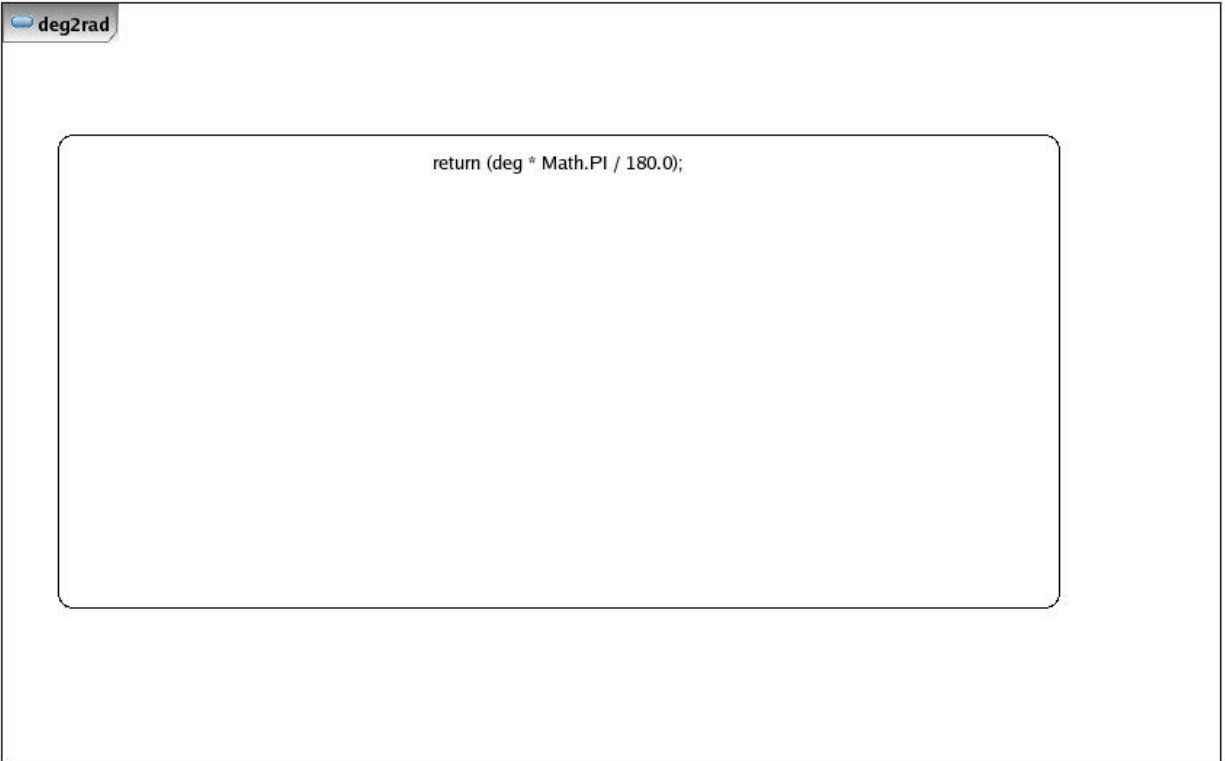
BDPkg::BDPkg::EventHandlerSM::Region1::JoinEvent::JoinEventEffect::Diagram1 -- Activity Diagram



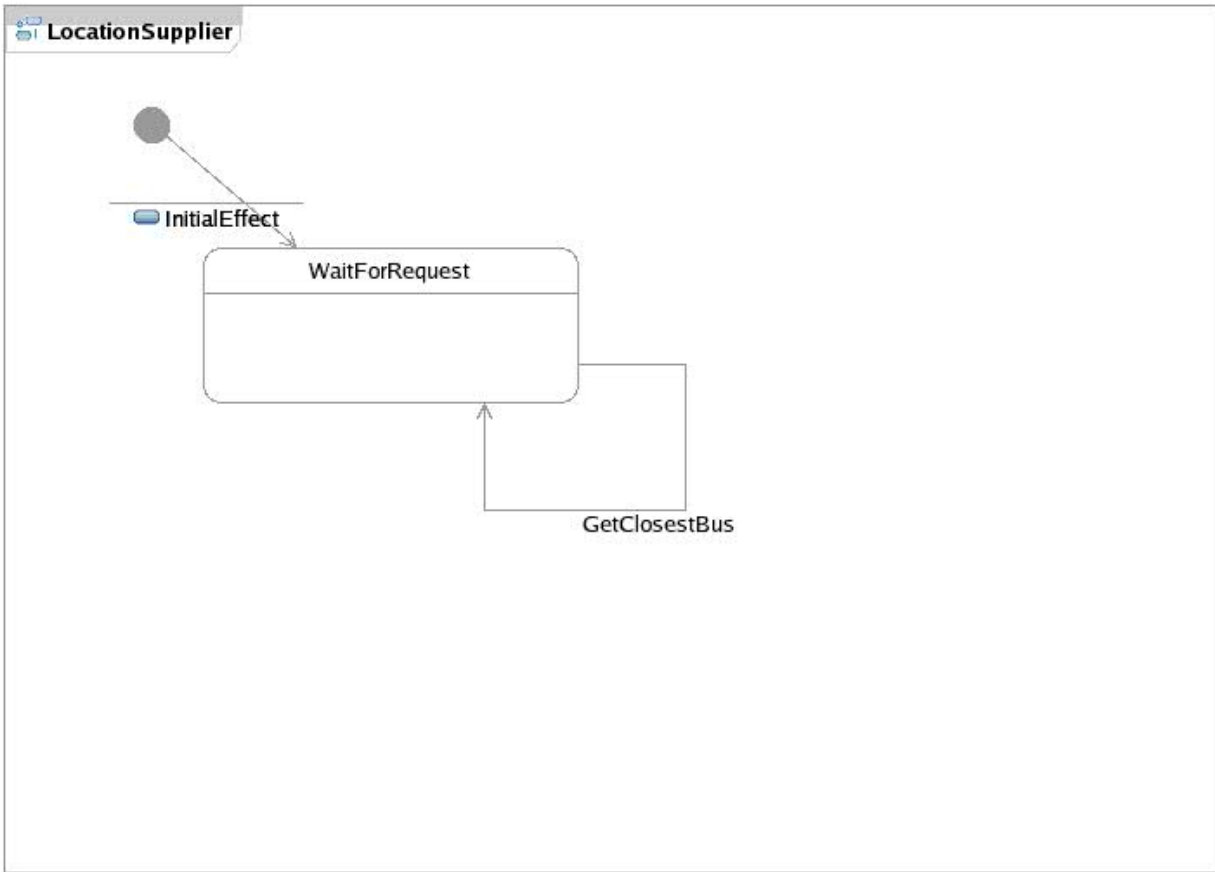
BDPkg::BDPkg::EventRouterMediator::forward::Diagram1 -- Activity Diagram



BDPkg::BDPkg::LocationSupplier::deg2rad::Diagram1 -- Activity Diagram



BDPkg::BDPkg::LocationSupplier::Diagram1 -- Statechart Diagram

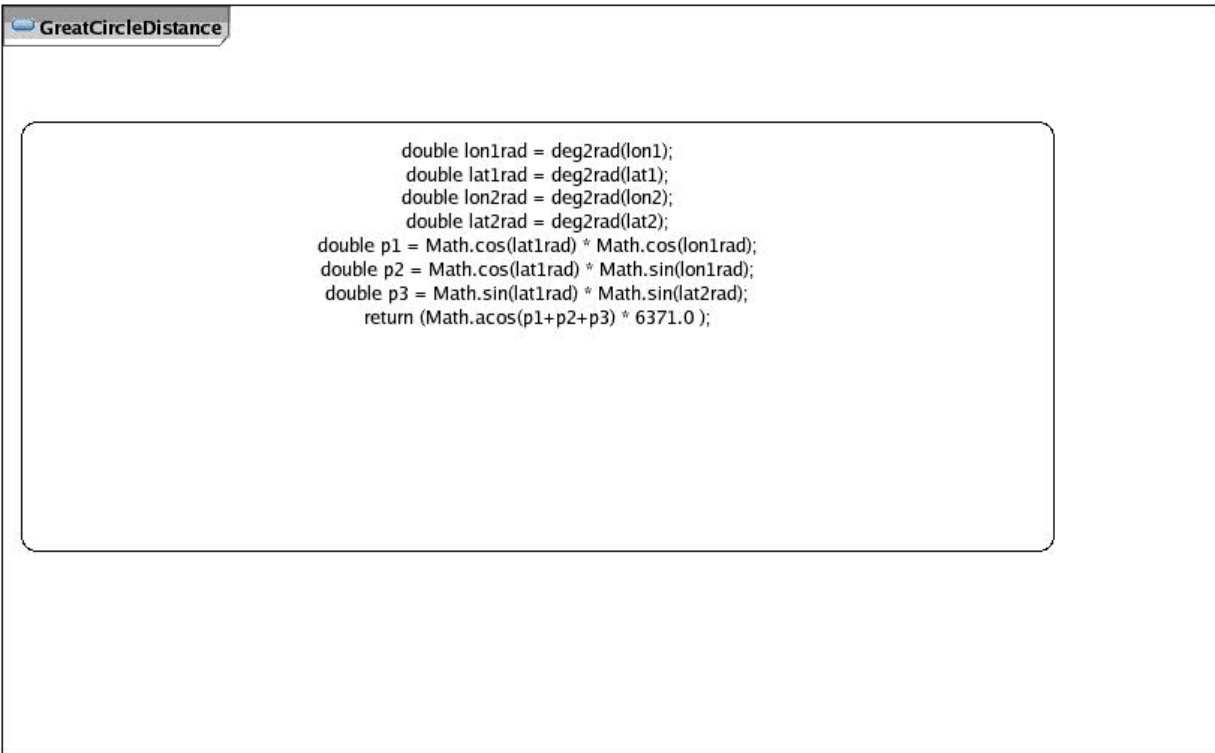


BDPkg::BDPkg::LocationSupplier::DMS2deg::Diagram1 -- Activity Diagram

DMS2deg

```
        //metode
        int lengde = dmsOrg.length();
        System.out.println("dmsOrg" + dmsOrg);
        String dms = dmsOrg.substring(1);
        String newDms = null;
        if (dms.startsWith("0")) {
            newDms = dms.substring(1);
        } else {
            newDms = dms.substring(0);
        }
        System.out.println("newDms" + newDms);
        double degrees = Integer.parseInt(newDms.substring(0,2));
        double minutes = Integer.parseInt(newDms.substring(2,4));
        double seconds = Integer.parseInt(newDms.substring(4,6));
        return (degrees + (minutes/60) + (seconds/3600));
```

BDPkg::BDPkg::LocationSupplier::GreatCircleDistance::Diagram1 -- Activity Diagram



BDPkg::BDPkg::LocationSupplier::Region1::<Transition>::InitialEffect::Diagram1 --  
Activity Diagram

```

        ArrayList l = new ArrayList();
        csm.busStopInfo = l;
        //skal lese inn alle stoppesteder for buss nr. 37 fra fil
        //må ha loop for å lese inn alle
        BusStop bs;
        String busStopName, bredde, lengde;
        int busStopId;
        BufferedReader inputfil = null;
        try
        {
            //BufferedReader inputfil = new BufferedReader
            //(new FileReader("/fi/fennis/h18/torunro/37inf5150NY.txt"));
            //BufferedReader inputfil = new BufferedReader (new FileReader("C:\\INF5150\\Drop2\\ZI\\37inf5150NY.txt"));
            try
            {
                //System.out.println("i try");
                // System.out.println("C:\\INF5150\\Drop 2\\ZIP\\37inf5150NY.txt");
                // inputfil = new BufferedReader (new FileReader("C:\\INF5150\\Drop 2\\ZIP\\37inf5150NY.txt"));
                // inputfil = new BufferedReader (new FileReader("C:/INF5150/Drop 2/ZIP/37inf5150NY.txt"));
                inputfil = new BufferedReader (new FileReader("/fi/fennis/h18/torunro/37inf5150NY1.txt"));
                //System.out.println("Filen er åpnet");
            }
            catch (FileNotFoundException filfeil)
            {System.out.println("Filen finnes ikke");}
            String record = inputfil.readLine();
            for (int i=1; i<30;i++)
            {
                busStopId = Integer.parseInt(record.substring(3,10));
                busStopName = record.substring(14,54);
                bredde = record.substring(77,84);
                lengde = record.substring(85,93);
                bs = new BusStop(busStopId, busStopName, bredde, lengde);
                boolean x = csm.busStopInfo.add(bs);
                record = inputfil.readLine();
            }
            //System.out.println("Etter for");
            busStopId = Integer.parseInt(record.substring(3,10));
            busStopName = record.substring(14,54);
            bredde = record.substring(77,84);
            lengde = record.substring(85,93);
            bs = new BusStop(busStopId, busStopName, bredde, lengde);
            boolean x = csm.busStopInfo.add(bs);
        }
        catch (IOException iofeil)
        {System.out.println("Feil ved lesing fra fil");}
        //catch (FileNotFoundException filfeil)
        //{System.out.println("Filen finnes ikke");}

```

BDPkg::BDPkg::LocationSupplier::Region1::GetClosestBus::GetClosestBusEffect::Diagram1  
 -- Activity Diagram

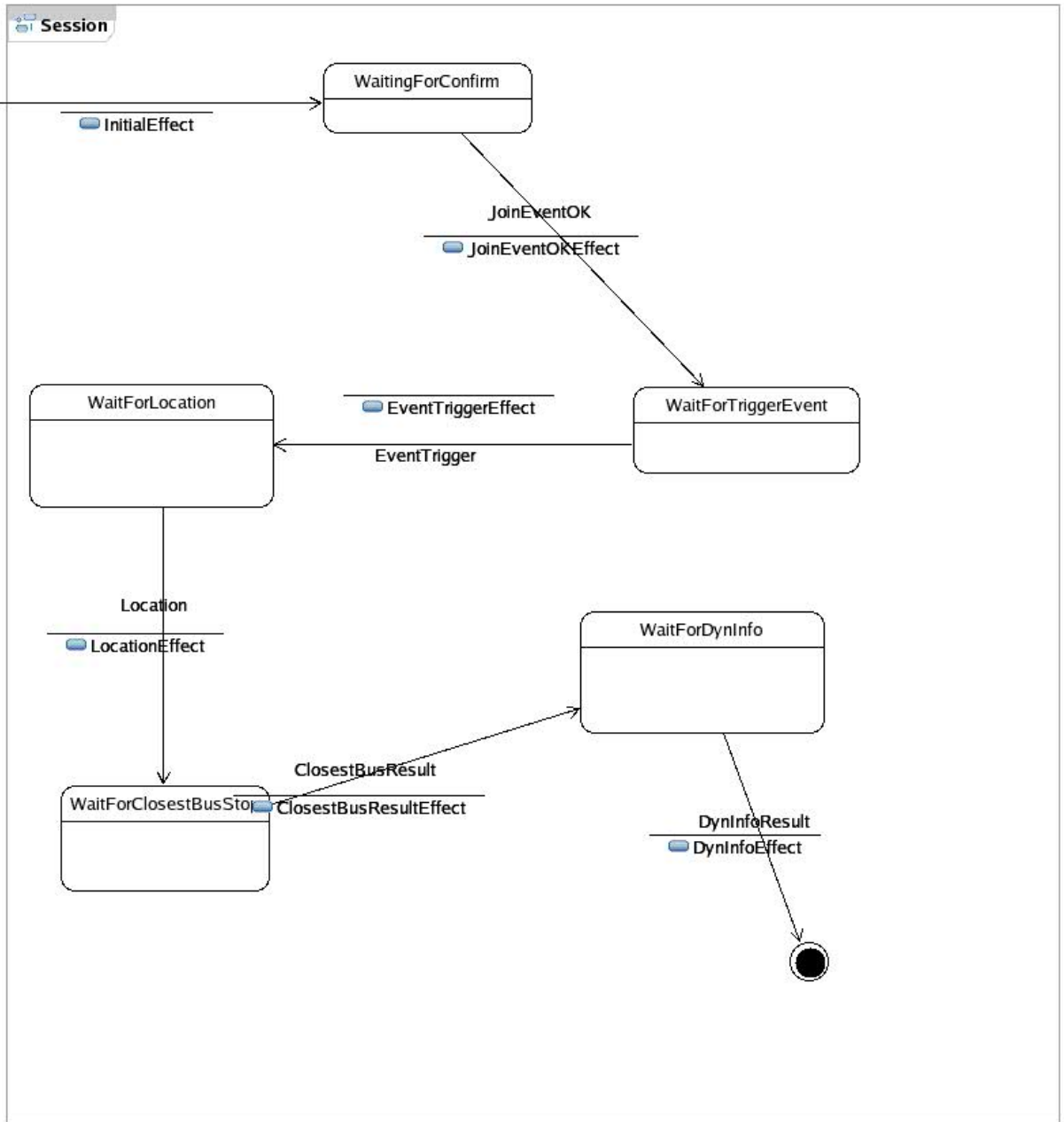
```
GetClosesBusEffect

System.out.println("CustPos: "+sig.custPos);
//variabler
String stopId = "";
double res1, min=0;
String stopName = ""; String bredde, lengde, b1, l1, b2, l2;
String mobBredde = sig.custPos.substring(0,7);
//System.out.println("mobBredde" + mobBredde);
String mobLengde = sig.custPos.substring(7,15);
//System.out.println("mobLengde" + mobLengde);
BusStop bp1, bp2;
boolean forsteGang = true;
for (int i=0; i<30; i++)
{
    bp1 = (BusStop) csm.busStopInfo.get(i);
    b1 = bp1.bredde;
    //System.out.println("bredde" + b1);
    //System.out.println("l:" + i);
    l1 = bp1.lengde;
    res1 = csm.GreatCircleDistance(csm.DMS2deg(mobBredde), csm.DMS2deg(mobLengde), csm.DMS2deg(b1), csm.DMS2deg(l1));
    if (forsteGang) {
        min = res1;
        stopId = "" + bp1.busStopId;
        stopName = bp1.busStopName;
        forsteGang = false; }
    else {
        if (res1<min)
        {min=res1;
        stopId = "" + bp1.busStopId;
        stopName = bp1.busStopName;}
    }}
    stopId = "SN$" + stopId;
output (new ClosestBusResult(sig.phoneNo, stopId, stopName, sig.konkat), csm.toController, csm);
```

BDPkg::BDPkg::Main -- Freeform Diagram

BDPkg::BDPkg::Session::Diagram1 -- Statechart Diagram



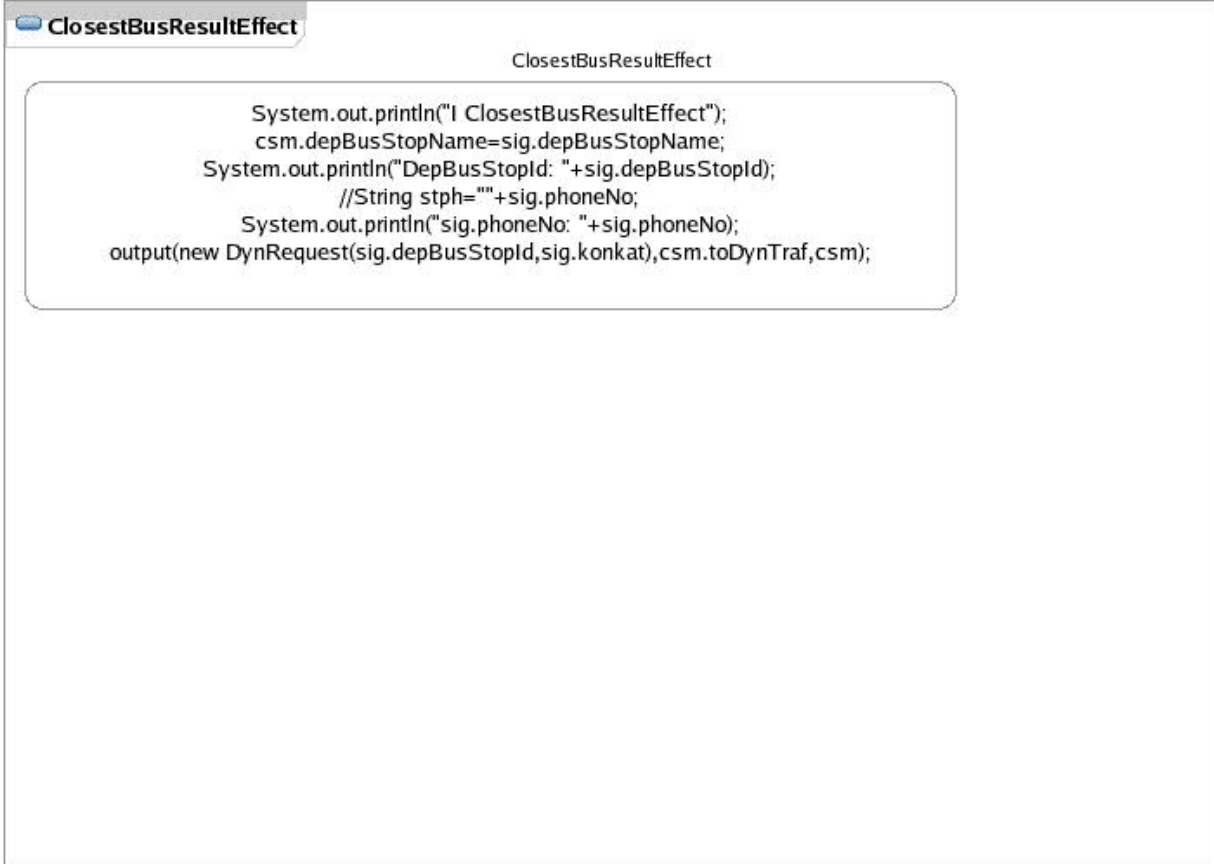


BDPkg::BDPkg::Session::Region1::<Transition>::InitialEffect::Diagram1 -- Activity Diagram

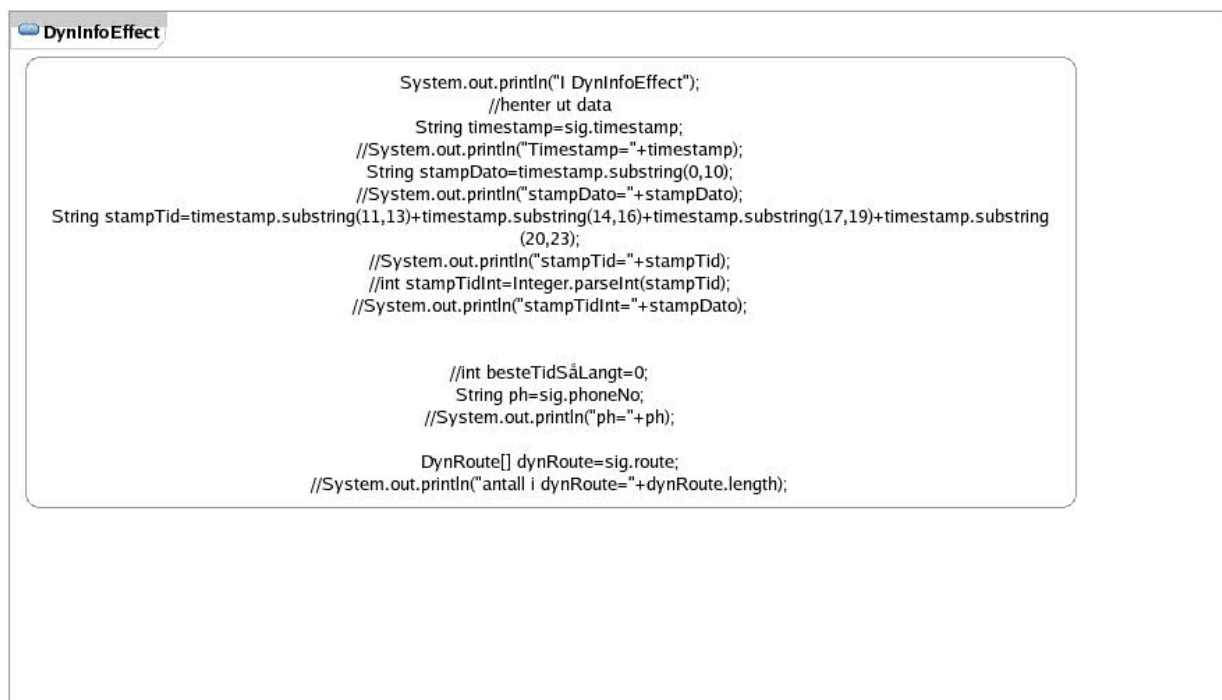
### InitialEffect

```
System.out.println("I session - setter attributter + trigger JoinEvent på eventHandler");
//sette attributter
PhoneNo ph=new PhoneNo(csm.p_ph);
csm.customerID=ph;
csm.eventStartTime=csm.p_time;
csm.eventType=csm.p_eventType;
csm.konkat_ph=csm.p_konkat_ph;
//trigge JoinEvent på eventHandler
output (new JoinEvent(csm.p_ph,csm.p_eventType,csm.p_time),csm.toEventH,csm);
```

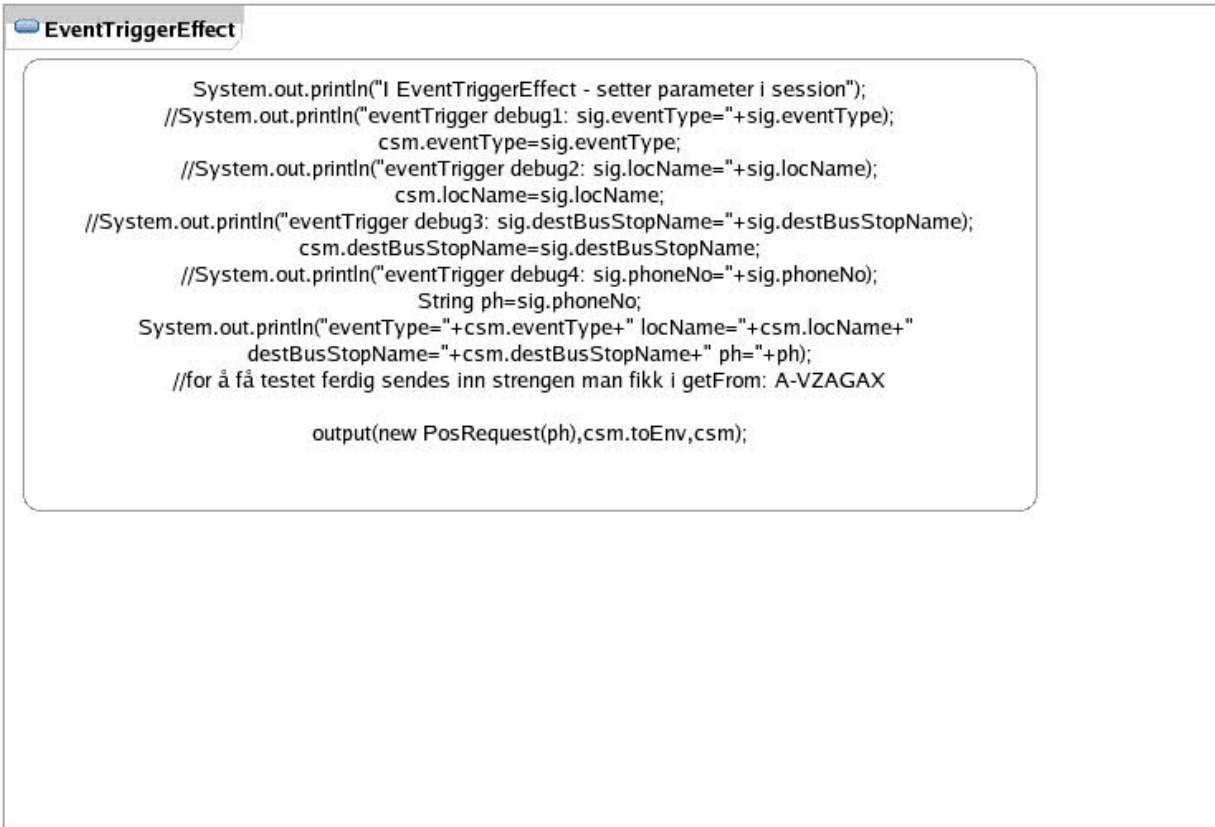
BDPkg::BDPkg::Session::Region1::ClosestBusResult::ClosestBusResultEffect::Diagram1  
-- Activity Diagram



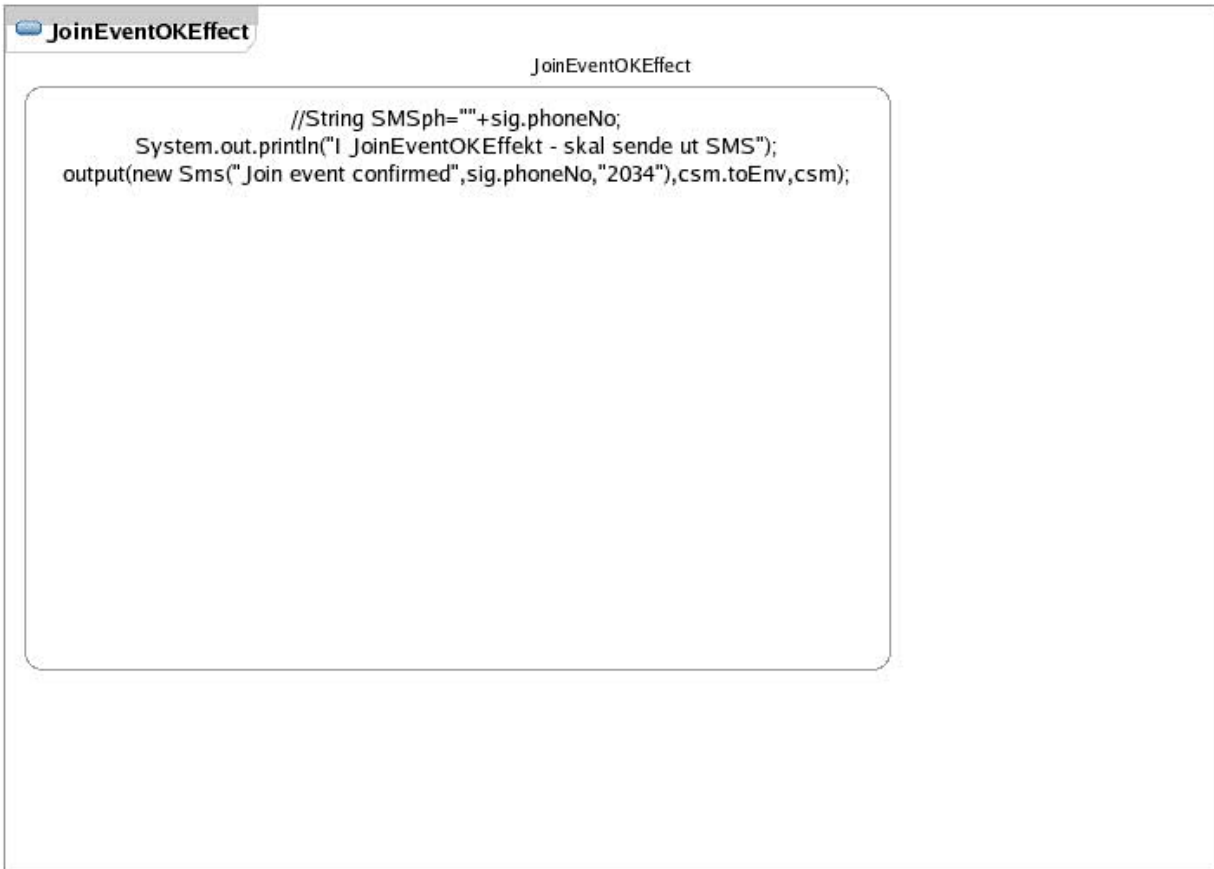
BDPkg::BDPkg::Session::Region1::DynInfoResult::DynInfoEffect::Diagram1 -- Activity Diagram



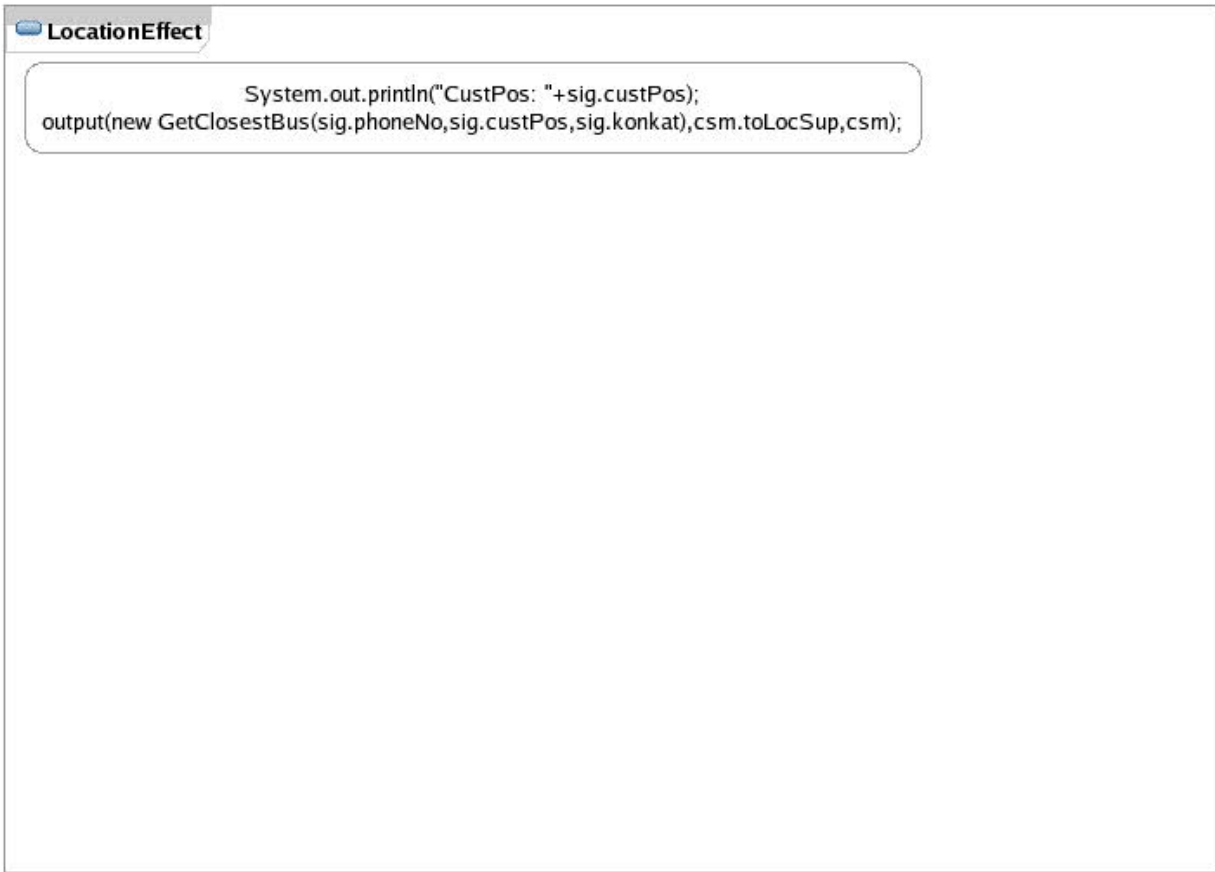
BDPkg::BDPkg::Session::Region1::EventTrigger::EventTriggerEffect::Diagram1 --  
Activity Diagram



BDPkg::BDPkg::Session::Region1::JoinEventOK::JoinEventOKEffect::Diagram1 --  
Activity Diagram



BDPkg::BDPkg::Session::Region1::Location::LocationEffect::Diagram1 -- Activity Diagram



BDPkg::BDPkg::SessionRouterMediator::forward::Diagram1 -- Activity Diagram

forward

forward

```
System.out.println("Router: mediatorList.size() = "+mediatorList.size());

for (int i=0;i<mediatorList.size();i++){
    ToSessionMediator mediator=(ToSessionMediator) mediatorList.get(i);
    System.out.println(mediator.phoneNo);

    if (sig instanceof JoinEventOK)
    {   System.out.println("joineventok");
        JoinEventOK signal=(JoinEventOK) sig;
        System.out.println("konkat="+signal.konkat);
        System.out.println("med="+mediator.phoneNo);
        if (signal.konkat.equals(mediator.phoneNo))
            { mediator.forward(sig); }
    }

    else if ( sig instanceof EventTrigger)
    {   System.out.println("eventtrigger");
        EventTrigger signal=(EventTrigger)sig;
        if (signal.konkat.equals(mediator.phoneNo))
            { mediator.forward(sig); }
    }

    else if ( sig instanceof Location)
    {   System.out.println("location");
        Location signal=(Location)sig;
        System.out.println("sig.phone"+signal.phoneNo);
        if (signal.konkat.equals(mediator.phoneNo))
            { System.out.println("er treff");
              mediator.forward(sig); }
    }

    else if ( sig instanceof ClosestBusResult)
    {   System.out.println("closestbusresult");
        ClosestBusResult signal=(ClosestBusResult)sig;
        if (signal.konkat.equals(mediator.phoneNo))
            { mediator.forward(sig); }
    }

    else if ( sig instanceof DynInfoResult)
    {   System.out.println("dyninforesult");
```

BDPkg::java::io::Main -- Freeform Diagram

BDPkg::java::Main -- Freeform Diagram

BDPkg::java::util::Main -- Freeform Diagram

BDPkg::Main -- Freeform Diagram



### **How we may understand the design as a refinement of the Drop 1 specification**

The semantics of state machines exclude inconclusive behaviour. Therefore all the inconclusive traces as presented in Drop1 become either positive or negative traces in the executable UML design of Drop1. By that we achieve refinement by supplementing.

By specifying concrete time constraints and concrete sets of events in the executable UML design of Drop1, we transfer all the other traces (which are beyond these constraints) into the set of negative traces. By doing so, we achieve the refinement by narrowing.

## Table of contents

### **1 Context identification**

- 1.1 BlindDateContext
- 1.2 BDSystem\_structure
- 1.3 RegisterCustomerSD
- 1.4 JoinEventSD
- 1.5 NotifyCustomersSD
- 1.6 targetOfEvaluationTable
- 1.7 valueDefTable
- 1.8 RiskValueFunc
- 1.9 RiskMatrix
- 1.10 AssetTable
- 1.11 riskEvaluationCriteriaTable

### **2 Risk identification**

- 2.1 HazOpTable
- 2.2 UnwantedIncidents2
- 2.3 UnwantedIncidents3
- 2.4 UnwantedIncidents1

### **3 Risk analysis**

- 3.1 ConFreqTable

### **4 Risk evaluation**

- 4.1 riskEvaTable
- 4.2 riskCatTable

### **5 Risk treatment**

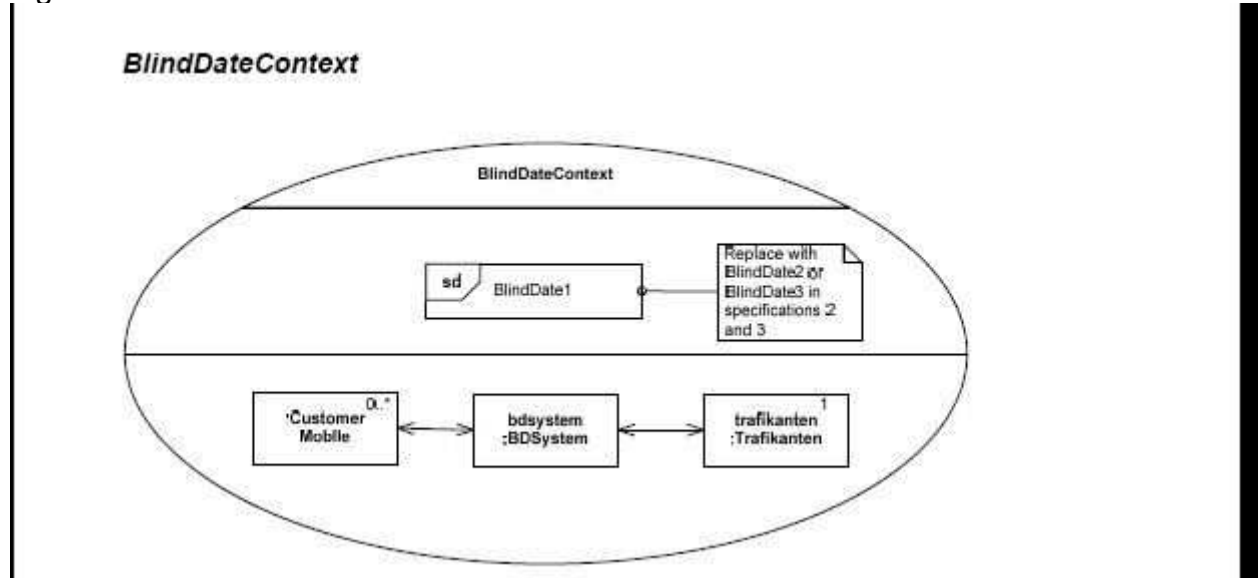
- 5.1 TreatmentIdentificationTable
- 5.2 RiskTreatment1
- 5.3 RiskTreatment2
- 5.4 TreatmentEvaluationTable

# 1 Context identificationError! Bookmark not defined.

## 1.1 BlindDateContext

Type: UML Model  
Name: BlindDateContext  
Short description: Blind Date System context diagram  
Concern: SWOT  
Viewpoint:  
Finalised:  
Full description: Blind Date System context diagram as shown in drop 1 solution.

Figure 1: BlindDateContext

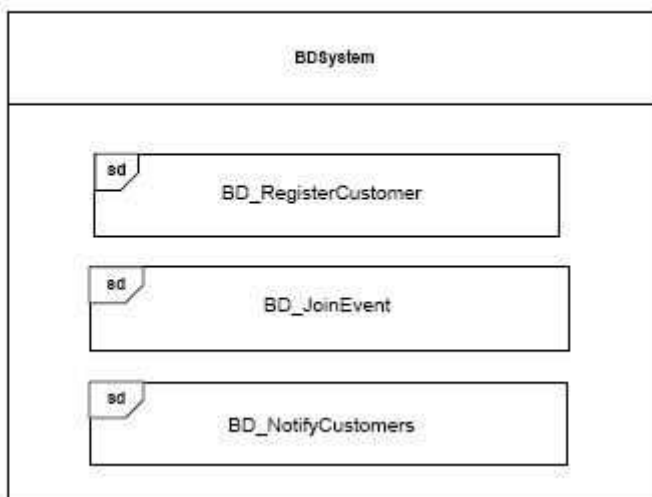


## 1.2 BDSysystem\_structure

Type: UML Model  
Name: BDSysystem\_structure  
Short description: BDSysystem structure diagram  
Concern: SWOT  
Viewpoint:  
Finalised:  
Full description: Structure diagram showing the main components of the Blind Date system.

Figure 2: BDSysystem\_structure

**BDSysystem structure**

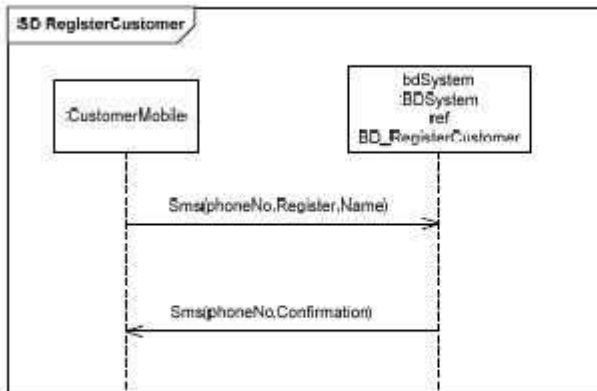


## 1.3 RegisterCustomerSD

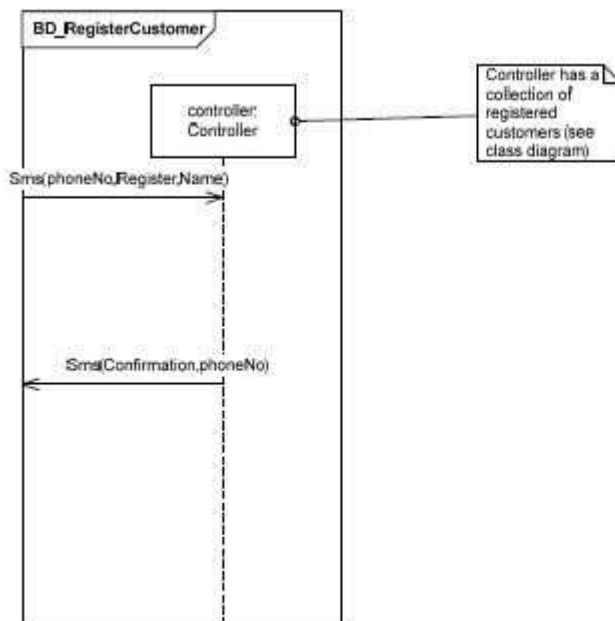
Type: UML Model  
 Name: RegisterCustomerSD  
 Short description: Register Customer sequence diagram  
 Concern: SWOT  
 Viewpoint:  
 Finalised:  
 Full description: Sequence diagram showing an instance of customer registration.

Figure 3: RegisterCustomerSD

RegisterCustomer sequence diagram



BD\_RegisterCustomer sequence diagram

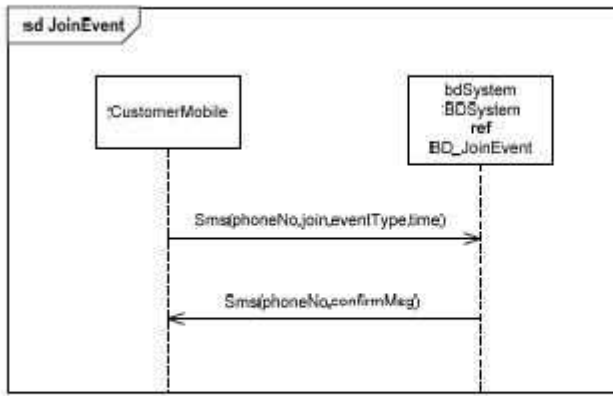


### 1.4 JoinEventSD

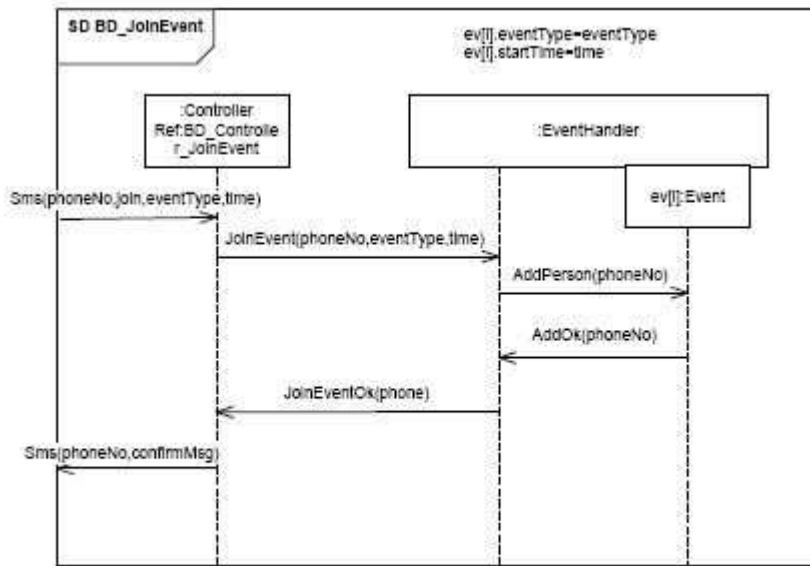
Type: UML Model  
 Name: JoinEventSD  
 Short description: Join Event sequence diagram  
 Concern: SWOT  
 Viewpoint:  
 Finalised:  
 Full description: Sequence diagram showing an instance of joining event.

Figure 4: JoinEventSD

## JoinEvent sequence diagram



## BD\_JoinEvent sequence diagram



### 1.5 NotifyCustomersSD

Type: UML Model

Name: NotifyCustomersSD

Short description: Notify customers system diagram

Concern: SWOT

Viewpoint:

Finalised:

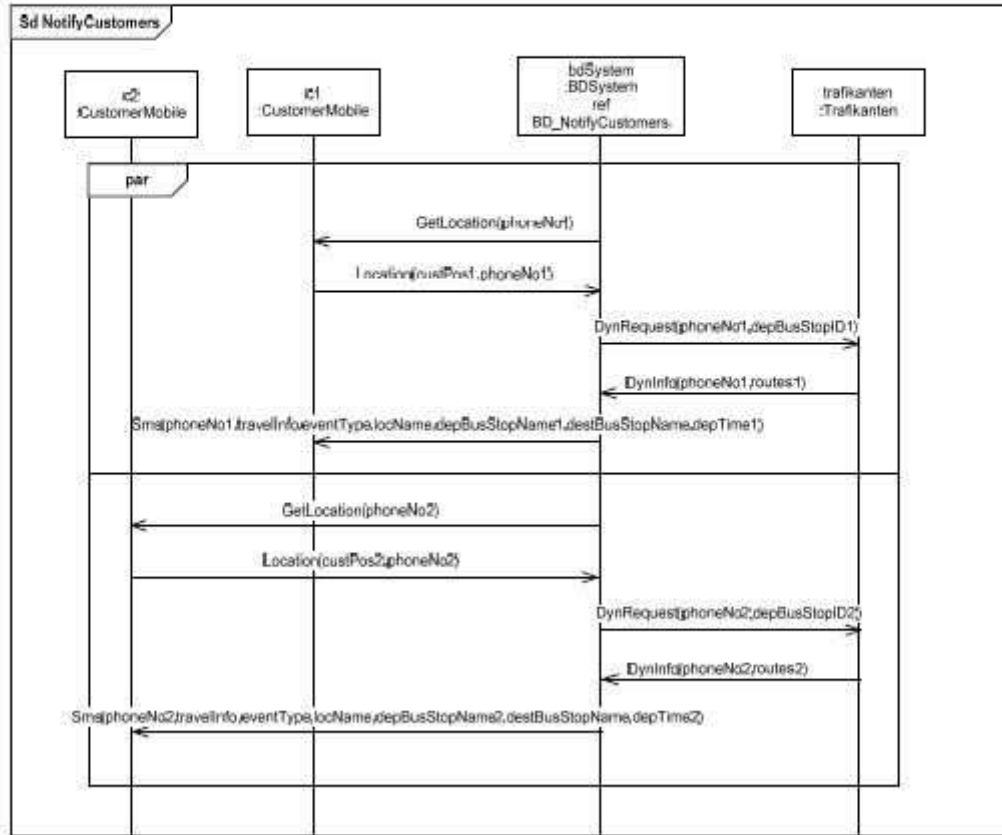
Full description: Diagram showing an instance of notifying a customer.

Figure 5: NotifyCustomersSD

## NotifyCustomers sequence diagram

At an appropriate time before an event the system collects information about the location for each participating customer, obtains travel information from Trafikanten, and sends travel advice to the customer.

The number of participants can differ between events; we have chosen to show the case with two participants for the same specific event. Note that the parameters eventTypes and locName are identical in the two operands of the par operator. This convention throughout the specification.



## 1.6 targetOfEvaluationTable

Type: Table  
 Name: targetOfEvaluationTable  
 Short description:  
 Concern: Target of evaluation  
 Viewpoint:  
 Finalised:  
 Full description: This tables describes the system under evaluation.

**Table 1: Target Of Evaluation Table**

Category	Value
Target	Blind Date System 1
Client	Blind Date Company
Service/Function	The system under analysis is described in "INF5150 Obligatory Exercise Drop 1 Proposed Solution October 21, 2005." The main components of the system are: --RegisterCustomer --JoinEvent --NotifyCustomer
Quality aspects	--Availability --Data Integrity --Confidentiality

## 1.7 valueDefTable

Type: Table  
 Name: valueDefTable  
 Short description: Value definition table for BD company  
 Concern: Target of evaluation  
 Viewpoint:  
 Finalised:  
 Full description: We assume that BD company has a yearly turnover of approx. 1m Nok, thus the monthly turnover of the company is approx. 80,000 Nok.

**Table 2: Value Definition Table**

Type	Domain	Allowed values	Description
Asset	Qualitative values	Low, Medium, High	Low: ~10K Nok Medium: ~80K Nok High: ~300K Nok
Frequency	Qualitative values	Rare, Unlikely, Possible, Likely, Certain	Rare: less than once per year. Unlikely: less than twice per year. Possible: about four times a year. Likely: ten times a year. Certain: each month.
Consequence	Qualitative values	Insignificant, Minor, Moderate, Major	Insignificant: no impact on business. Minor delays. Minor: loss of profits. Moderate: loss of clients. Major: loss of business sector, close to bankruptcy.
Risk value	Qualitative values	Low, Moderate, Major	Low: 'rare' to 'possible' frequency, 'incignificant' to 'moderate' consequence. Moderate: 'rare' to 'certain' frequency, 'insignificant' to 'major' consequence. Major: 'possible' to 'certain' frequency, 'minor' to 'major' consequence.

**1.8 RiskValueFunc**

Type: Table  
Name: RiskValueFunc  
Short description: Risk value function  
Concern: Target of evaluation  
Viewpoint:  
Finalised:  
Full description:

**Table 3: Risk Value Function**

Risk value function
We are using qualitative values for frequency and concequence, therefore the risk value function cannot be defined explicitly.

**1.9 RiskMatrix**

Type: Table  
Name: RiskMatrix  
Short description: Risk matrix for DB company  
Concern: Target of evaluation  
Viewpoint:  
Finalised:  
Full description:

**Table 4: Risk Matrix**

Frequency	Insignificant	Minor	Moderate	Major
Rare	Low	Low	Low	Moderate
Unlikely	Low	Low	Low	Moderate
Possible	Low	Major	Moderate	Major
Likely	Moderate	Major	Major	Major
Certain	Moderate	Major	Major	Major

**1.10 AssetTable**

Type: Table  
Name: AssetTable  
Short description: Asset table for BD company  
Concern: Assets  
Viewpoint:  
Finalised:  
Full description: The assets of the BD company.

**Table 5: Asset Table**

Asset ID	Description	Category	Value
A1	Customers	Human	High
A2	JoinEvent-list	Information	Medium

Asset ID	Description	Category	Value
A3	Data Equipment	Physical	Low
A4	Event Register	Information	Medium
A5	Income	Other	High
A6	Reputation	Human	High
A7	System design and source code	Information	High

### 1.11 riskEvaluationCriteriaTable

Type: Table  
Name: riskEvaluationCriteriaTable  
Short description: Risk evaluation criteria for BD company  
Concern: Risk evaluation criteria  
Viewpoint:  
Finalised:  
Full description:

**Table 6: Risk Evaluation Criteria Table**

Criteria ID	Criteria	Description	Applied for assets
C1	Low	If "Risk level" is equal to "Low" then "Accept the risk"	A3
C2	Moderate	If "Risk level" is equal to "Moderate" then "Monitor the risk"	A7, A2, A4
C3	Major	If "Risk level" is equal to "Major" then "Treat the risk"	A6, A5, A1

## 2 Risk identificationError! Bookmark not defined.

### 2.1 HazOpTable

Type: Table  
Name: HazOpTable  
Short description: HazOP table for BD company  
Concern: Threats  
Viewpoint:  
Finalised:  
Full description:

**Table 7: HazOp Table**

HazOp ID	Asset ID	Reference	Guideword	Threat	Incident	Scenario
1	A1	RegisterCustomer	Unavailable	Human error of the developer	Customer is not registered in the database	Confirmation received by the customer, but the customer is not registered
2	A2	JoinEvent	Integrity	Hacker	Loss of JoinEvent-list, faulty "event not found" message to the customers who try to sign up	Intrusion into the system, deletion of the database, containing the event-list
3	A6	JoinEvent	Unintentional	Human error of the developer	Customer turns up at a wrong event	Customer added to the wrong event
4	A4	EventRegister	Unavailable	Hacker, Data Equipment	Customers cannot join events or register	Blind date system out of function
5	A7	BlindDateContext	Deliberate	Unfaithful employee	Market advantage is lost	Design disclosed to competitor
6	A5	NotifyCustomer	Unavailable	Data Equipment	Customers do not come to the event	Trafikanten or/and SMS is not functioning and the customers thus do not receive notification
7	A6	NotifyCustomer	Delay	Data Equipment	Customer is late for the event or	SMS is not delivered in time

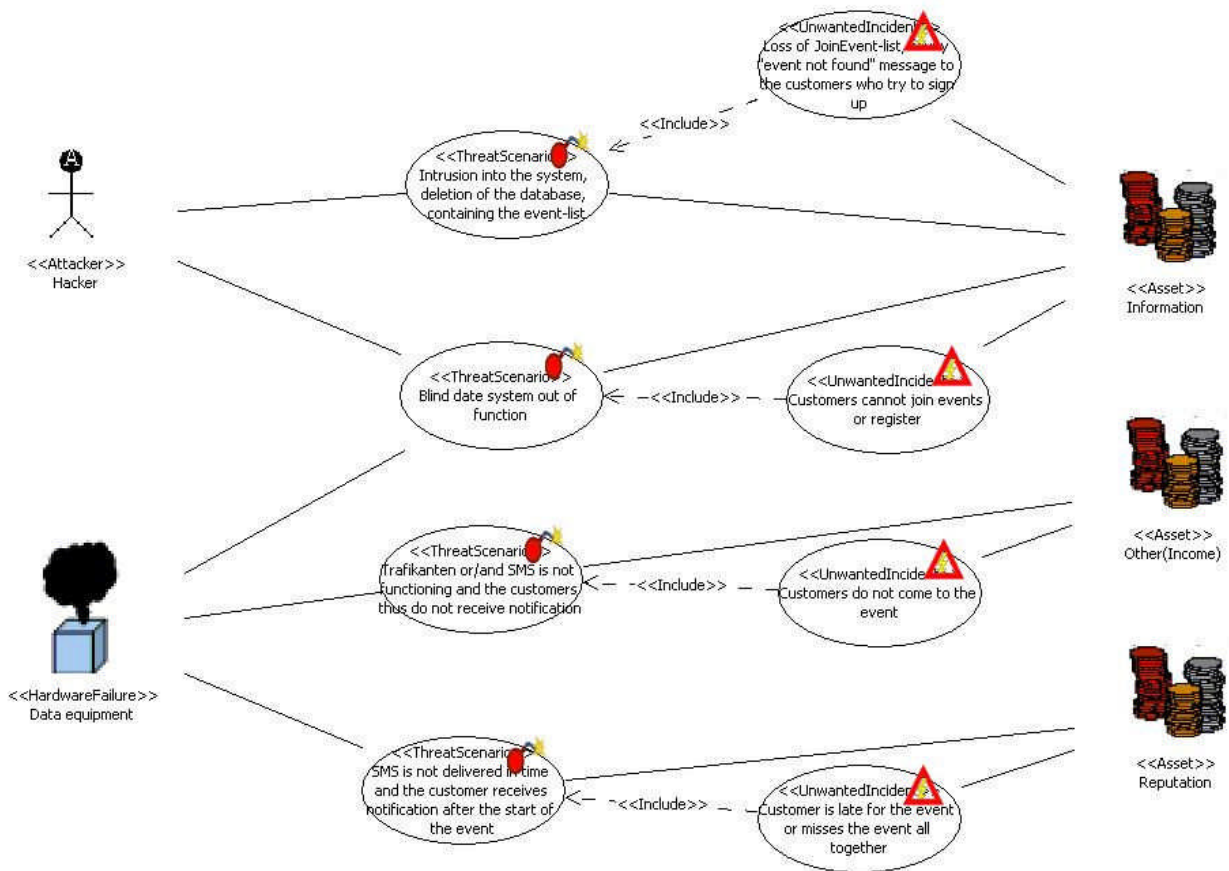


HazOp ID	Asset ID	Reference	Guideword	Threat	Incident	Scenario
					misses the event all together	and the customer receives notification after the start of the event

## 2.2 UnwantedIncidents2

Type: UML Model  
Name: UnwantedIncidents2  
Short description: Unwanted Incidents for BD company  
Concern: Threats  
Viewpoint:  
Finalised:  
Full description: Unwanted incidents for involving hacker and data equipment.

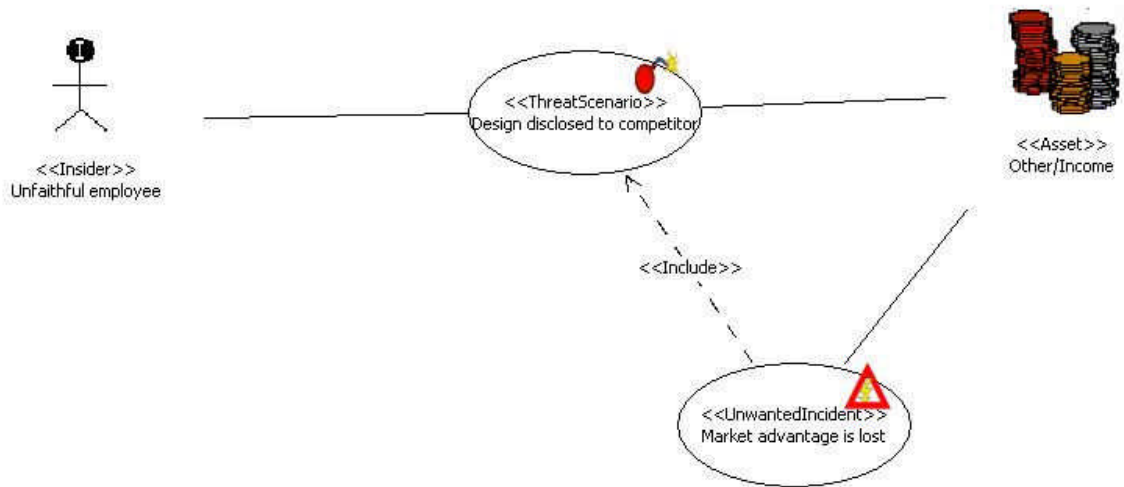
Figure 6: UnwantedIncidents2



## 2.3 UnwantedIncidents3

Type: UML Model  
Name: UnwantedIncidents3  
Short description: Unwanted Incidents for BD company  
Concern: Threats  
Viewpoint:  
Finalised:  
Full description: Unwanted incidents related to unfaithful employee

Figure 7: UnwantedIncidents3



## 2.4 UnwantedIncidents1

Type: UML Model

Name: UnwantedIncidents1

Short description: Unwanted Incidents for BD company

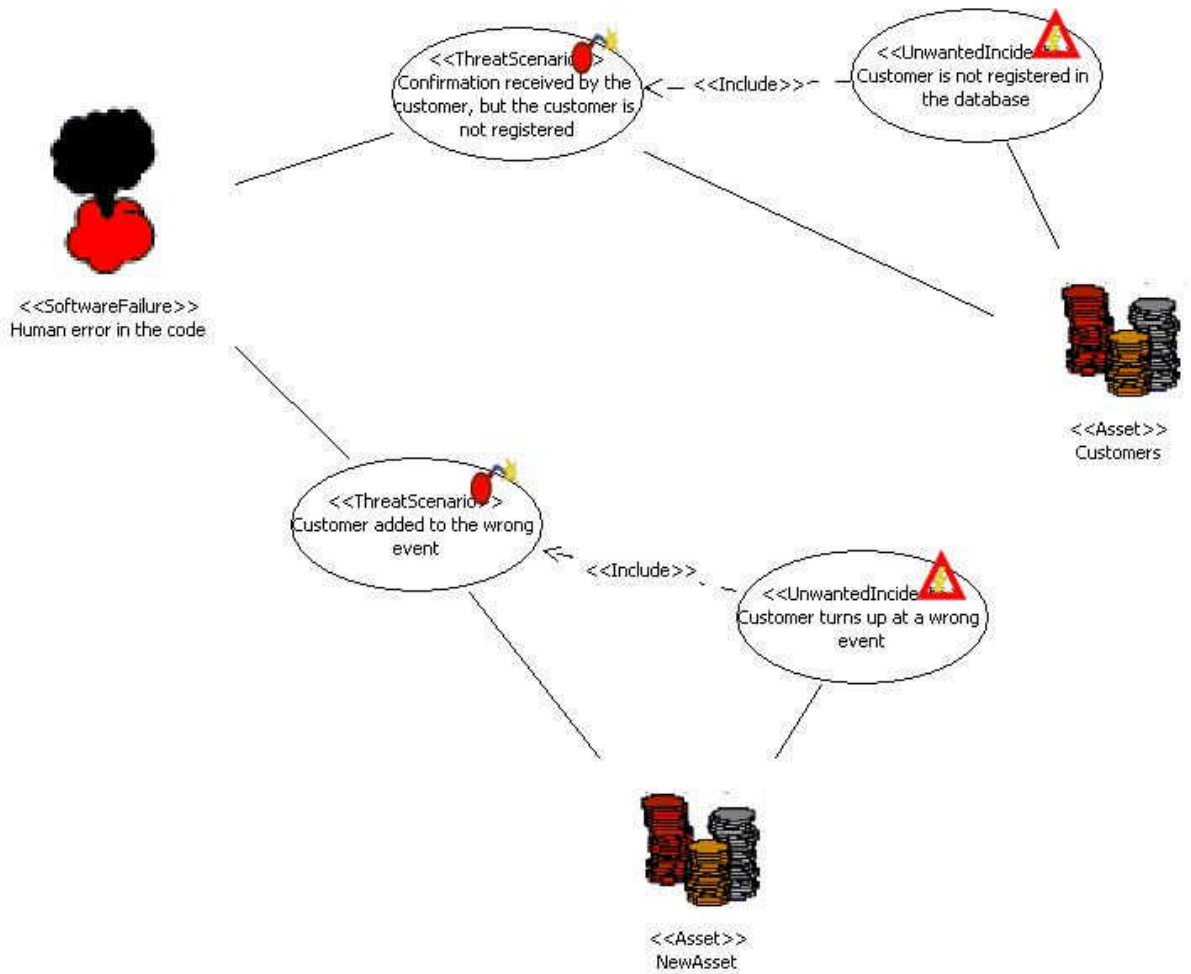
Concern: Unwanted incidents

Viewpoint:

Finalised:

Full description: Unwanted incidents related to human error of the developer

Figure 8: UnwantedIncidents1



### 3 Risk analysis Error! Bookmark not defined.

#### 3.1 ConFreqTable

Type: Table  
 Name: ConFreqTable  
 Short description: Consequences and Frequency table for BD system  
 Concern: Consequence  
 Viewpoint:  
 Finalised:  
 Full description:

**Table 8: Consequence and Frequency Table**

Risk ID	Asset ID	Incident	Consequence Value	Frequency Value
R1	A1	Customer is not	Moderate	Unlikely

Risk ID	Asset ID	Incident	Consequence Value	Frequency Value
		registered in the database		
R2	A2	Loss of JoinEvent-list, faulty "event not found" message to the customers who try to sign up	Major	Possible
R3	A6	Customer turns up at a wrong event	Moderate	Possible
R4	A4	Customers cannot join events or register	Major	Unlikely
R5	A7	Market advantage is lost	Major	Unlikely
R6	A5	Customers do not come to the event	Major	Likely
R7	A6	Customer is late for the event or misses the event all together	Minor	Likely

## 4 Risk evaluationError! Bookmark not defined.

### 4.1 riskEvaTable

Type: Table

Name: riskEvaTable

Short description: Risk Evaluation table for BD company

Concern: Risk estimates

Viewpoint:

Finalised:

Full description: 1--is top priority (treat the risk). 2--is medium priority (risk can be monitored). 3--is low priority (risk can be ignored).

**Table 9: Risk Evaluation Table**

Risk ID	Risk Value	Risk Priority
R1	Low	2
R2	Major	1
R3	Moderate	2
R4	Moderate	2
R5	Moderate	2
R6	Major	1
R7	Major	3

### 4.2 riskCatTable

Type: Table

Name: riskCatTable

Short description: Risk category table for BD company

Concern: Risk estimates

Viewpoint:

Finalised:

Full description: Cat1--risks associated with human error of developer. Cat2--risks associated with malicious intrusion. Cat3--risks associated with stealing of the program design. Cat4--risks associated with data equipment. Cat5--risks associated with subproviders (GSM network providers, Trafikanten).

**Table 10: Risk Category Table**

Risk category ID	Risks
Cat1	R1, R3
Cat2	R2, R4
Cat3	R5
Cat4	R4
Cat5	R6, R7

## 5 Risk treatmentError! Bookmark not defined.

## 5.1 TreatmentIdentificationTable

Type: Table  
Name: TreatmentIdentificationTable  
Short description: Treatment Identification table for BD company  
Concern: Treatment  
Viewpoint:  
Finalised:  
Full description:

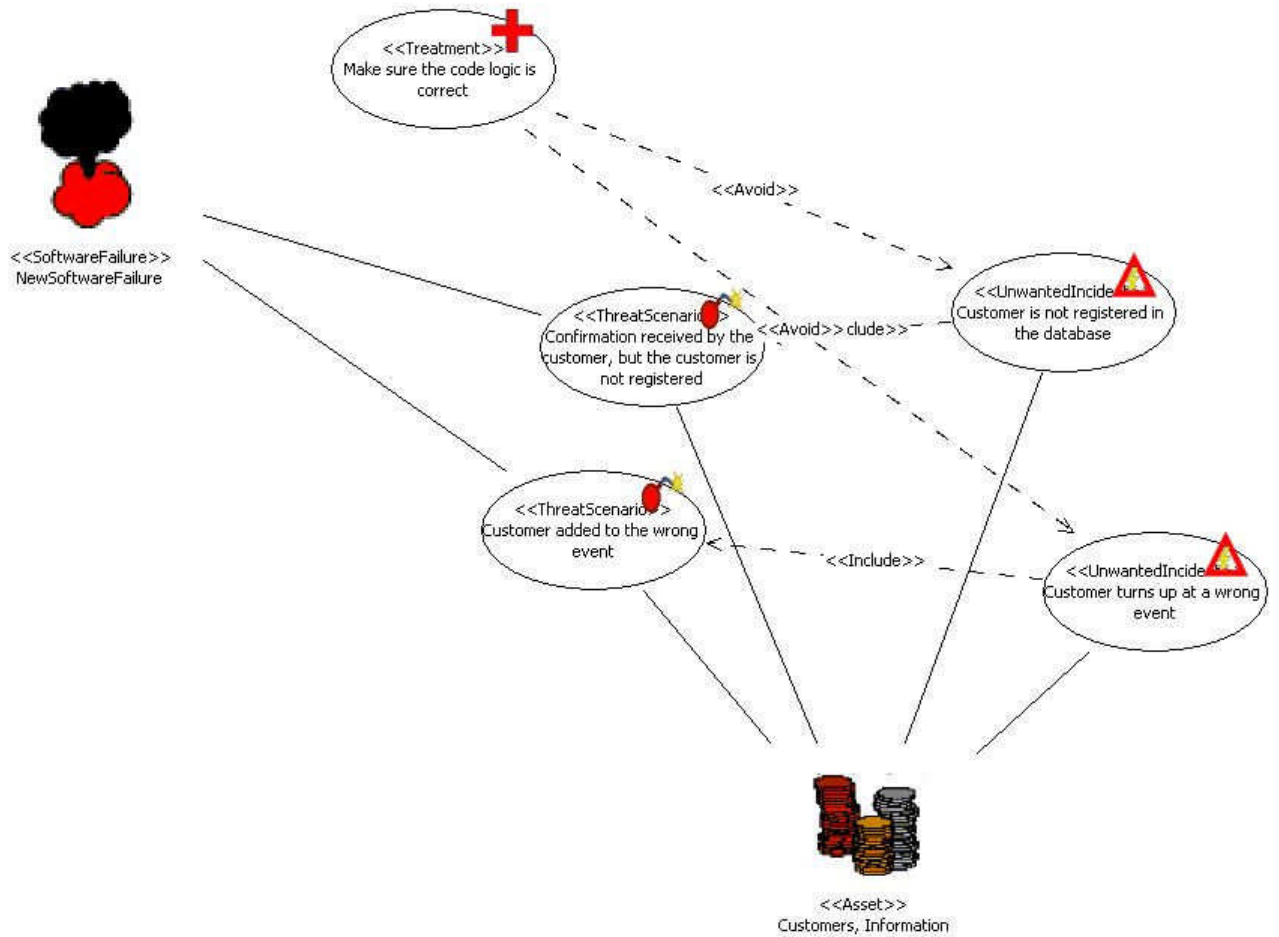
**Table 11: Treatment Identification Table**

Treatment ID	Risk ID/category	Treatment strategy	Description	References
TR1	Cat1	Avoid	Make sure the code logic is correct	
TR2	Cat2	Transfer	Outsource the database to a secured provider	
TR3	Cat1	Avoid	Make sure the code logic is correct	
TR4	Cat4	Transfer	Outsource the system hardware to a provider with backup solutions	
TR5	Cat3	Retain	Security measures maybe a way too expensive	
TR6	Cat5	Avoid	Make sure backup solutions are in place for GSM	
TR7	Cat5	Retain	Unpredictable event	

## 5.2 RiskTreatment1

Type: UML Model  
Name: RiskTreatment1  
Short description: RiskTreatment for BD company  
Concern: Treatment  
Viewpoint:  
Finalised:  
Full description: Risk treatment suggesting correction of code logic.

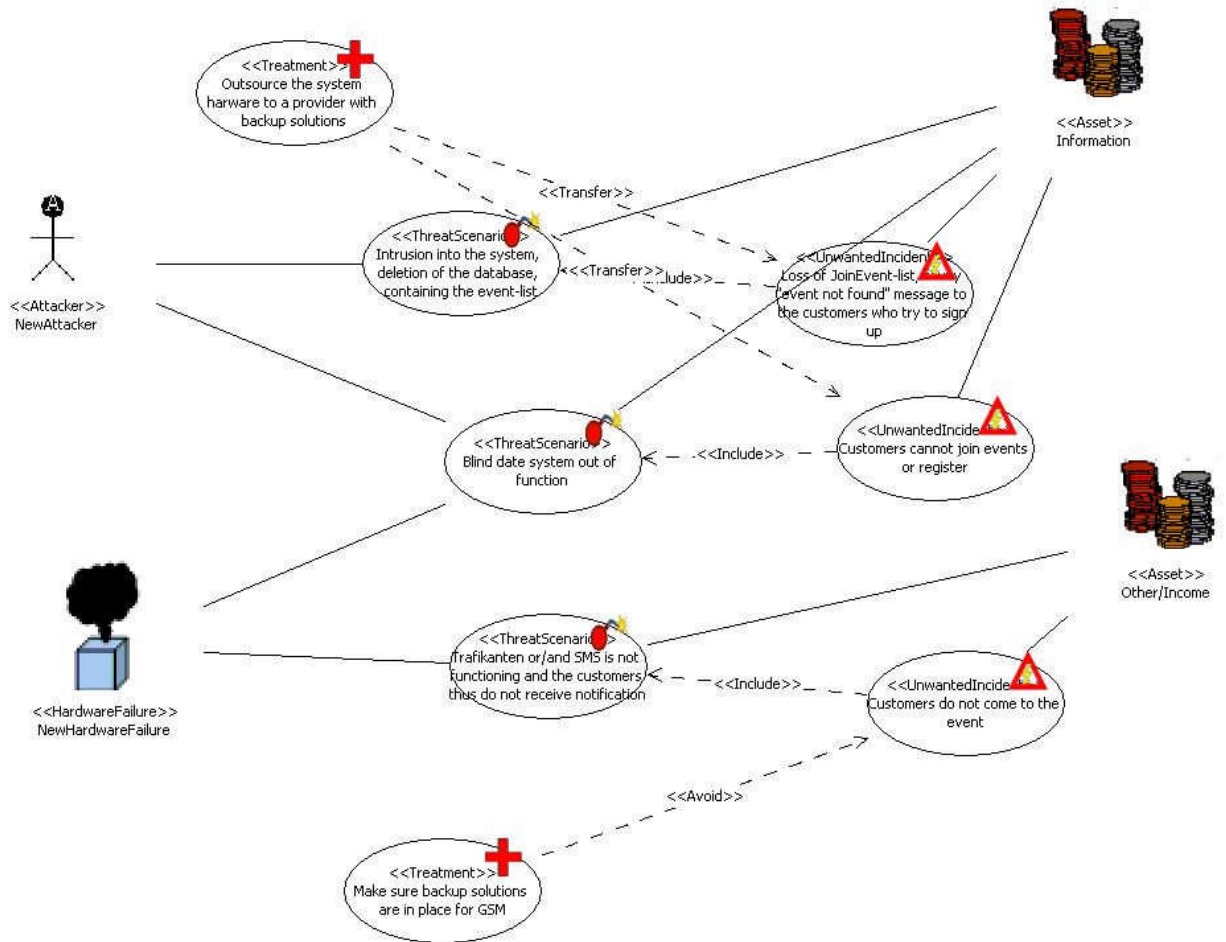
Figure 9: RiskTreatment1



### 5.3 RiskTreatment2

Type: UML Model  
 Name: RiskTreatment2  
 Short description: Risk treatment for BD company  
 Concern: Treatment  
 Viewpoint:  
 Finalised:  
 Full description: Risk treatment suggesting outsourcing and backup.

Figure 10: RiskTreatment2



## 5.4 TreatmentEvaluationTable

Type: Table  
 Name: TreatmentEvaluationTable  
 Short description: Treatment Evaluation table for BD company  
 Concern: Treatment assessment  
 Viewpoint:  
 Finalised:  
 Full description: 1--is top priority. 2--is medium priority. 3--is low priority.

**Table 12: Treatment Evaluation Table**

Treatment ID	Risk reduction	Criteria met?	Benefit	Cost	Priority
TR1	Risk level: medium	Partly	Reduced logic errors in the system	10 hours work = 10,000 Nok	2
TR2	Risk level: low	Yes	Secured database	1000 Nok a month	1
TR3	Risk level: medium	Partly	Reduced logic errors in the system	10 hours work = 10,000 Nok	2
TR4	Risk level: low	Yes	Secured hardware	1000 Nok a month	1
TR5	Risk level: as it was	No	No expenses for security	0 Nok a month	3
TR6	Risk level: medium	Partly	Not dependant on one GSM provider	Extra subscription fee to another GSM provider	2
TR7	Risk level: as it was	No	No extra expenses	0 Nok	3