Adriano FRANCI
Carlos De sa Matos
Sebastient Valenne

HWPS , Heat Wave Prevention System
Messir Analysis Document
- v 1.0 -

(*Report type: Default*)

Friday 16th December, 2016 - 13:16

Contents

List of Figures

Listings

Chapter 1

Introduction

1.1 Overview

1.2 Purpose and recipients of the document

1.3 Application Domain

1.4 Definitions, acronyms and abbreviations

1.5 Document structure

Chapter 2

General Description

2.1 Domain Stakeholders

2.2 System's Actors

The objective of this section is not to provide the full requirement elicitation document in this section but to reuse a part of this document to provide a informal introduction to the Messir specification of the system under development. The use case model is made of a use case diagrams modelling abstractly and informally the actors and their use cases together with a set of use cases descriptions. In addition, those diagrams and description tables are adapted to the Messir specification since actor and messages names together with parameters are partly adapted to be consistent with the specification identifiers (see [1] for more details).

2.3 Use Cases Model

This section contains the use cases elicited during the requirements elicitation phase. The use cases are textually described as suggested by the Messir method and inspired by the standard Cokburn template [2].

2.3.1 Use Cases

2.3.1.1 summary-suAcceptMission

The actVolunteer goal is to accept a HelpRequest mission and be assigned to it

Use-Case Description	
Name	suAcceptMission
Scope	system
Level	summary
Primary actor(s)	
1	`actVolunteer[active]`
Goal(s) description	
The actVolunteer goal is to accept a HelpRequest mission and be assigned to it	
Protocol condition(s)	
1	The system is deployed
2	The actVolunteer is authorized by the Coordinator to accept missions
Pre-condition(s)	
1	
Main post-condition(s)	
1	The HelpRequest has been assigned to the actVolunteer and actVolunteer has been informed of the success of the operation
Main Steps	
a	the actor `actVolunteer` executes the `ugGetMissionInRange` use case
b	the actor `actVolunteer` executes the `oeAcceptMission` use case
Additional Information	
none	

2.3.1.2 summary-suAltertAFamilyMember

Figure 2.1 shows use case view for Alert a family member summary use case view

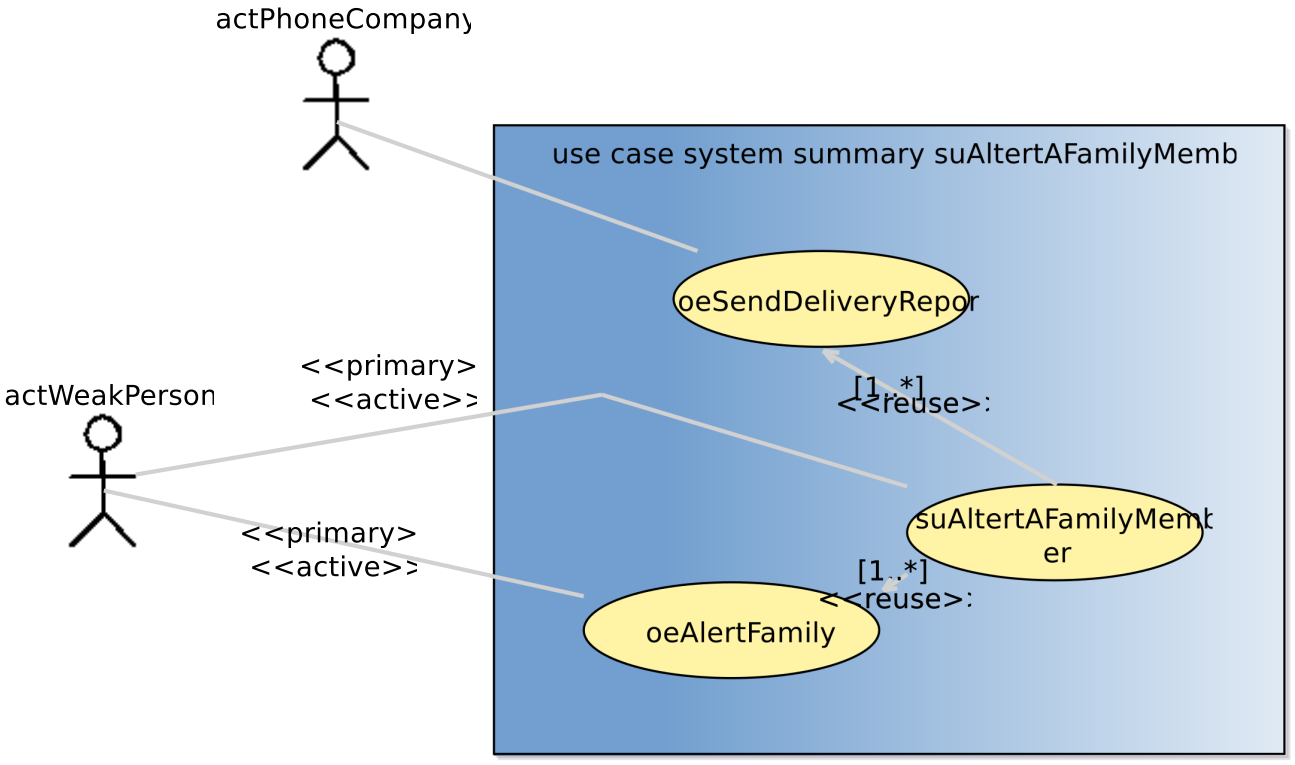


Figure 2.1: suAlertAFamilyMemeber summary use case view

2.3.1.3 summary-suCallSelectedHelpRequest

The goal is to successfully call the weak person by choosing the help request it sent

USE-CASE DESCRIPTION	
Name	suCallSelectedHelpRequest
Scope	system
Level	summary
Primary actor(s)	
1	`actCoordinator[active]`
Secondary actor(s)	
1	`actPhoneCompany[]`
Goal(s) description	
The goal is to successfully call the weak person by choosing the help request it sent	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request still exists and is still pending in the list
Main post-condition(s)	
1	The coordinator is linked to the weak person via phone call
Main Steps	
a	the actor `actCoordinator` executes the `oeReqCall` use case
b	the actor `actPhoneCompany` executes the `oeGetConfirm` use case

continues in next page . . .

. . . Use-Case Description table continuation

Additional Information
none

Figure 2.2 Use case view for call the selected help request

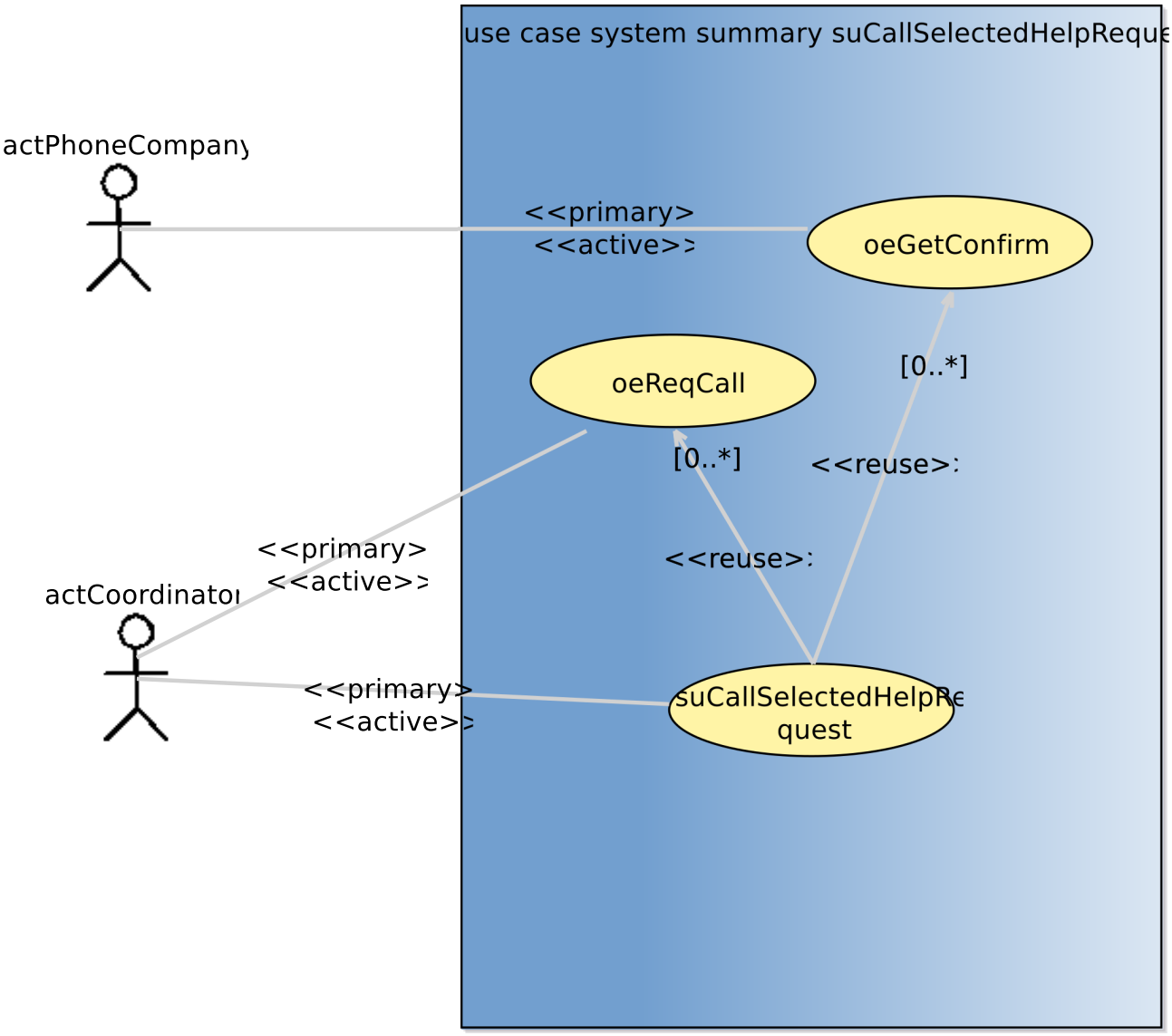


Figure 2.2:

2.3.1.4 summary-suDeployRun

This represent the normal run of the program including a weak person requesting help, coordinator retrieving help request ,calling it, assigning

Use-Case Description	
Name	suDeployRun
Scope	system
Level	summary
Primary actor(s)	
1	actCoordinator[active]
2	actVolunteer[active]
3	actWeakPerson[active]
4	actActivator[proactive]
Goal(s) description	
This represent the normal run of the program including a weak person requesting help, coordinator retrieving help request ,calling it, assigning	
Protocol condition(s)	
1	
Pre-condition(s)	
1	
Main post-condition(s)	
1	
Main Steps	
a	the actor actWeakPerson executes the ugRequestHelp use case
b	the actor actCoordinator executes the ugRetrievePendingHelpRequestDetails use case
c	the actor actCoordinator executes the suCallSelectedHelpRequest use case
d	the actor actCoordinator executes the ugAssignPriorityToHelpRequest use case
e	the actor actActivator executes the oeSetClock use case
f	the actor actActivator executes the oeSendNotificationToVolunteer use case
g	the actor actVolunteer executes the ugGetMissionInRange use case
h	the actor actVolunteer executes the suRetrieveMissionDetails use case
i	the actor actVolunteer executes the suAcceptMission use case
Additional Information	
none	

Figure 2.3 shows the use case diagram for the suDeployRun summary use case

2.3.1.5 summary-suRetrieveMissionDetails

This Use case summary allows the volunteer to retrieve a specific Help request detail

Use-Case Description	
Name	suRetrieveMissionDetails
Scope	system
Level	summary
Primary actor(s)	
1	actVolunteer[active]
Goal(s) description	
This Use case summary allows the volunteer to retrieve a specific Help request detail	
Protocol condition(s)	

continues in next page . . .

. . . Use-Case Description table continuation

1	The System is deployed
2	The volunteer is logged in and has executed getMission in at least one
Pre-condition(s)	
1	PendingMission list is not null
Main post-condition(s)	
1	The helpRequest (mission) details are returned to volunteer
Main Steps	
a	the actor `actVolunteer` executes the `ugGetMissionInRange` use case
b	the actor `actVolunteer` executes the `oeGetMissionDetails` use case
Additional Information	
none	

Figure 2.4 shows the use case diagram for the suRetrieveMissionDetails summary use case

2.3.1.6 summary-ugRequestHelp

Figure 2.5 shows the use case diagram for the ugRequestHelp usergoal use case

2.3.1.7 usergoal-ugAssignPriorityToHelpRequest

The goal is to assign a priority to a help request after a call has been made with the weak person

USE-CASE DESCRIPTION	
Name	ugAssignPriorityToHelpRequest
Scope	system
Level	usergoal
Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is to assign a priority to a help request after a call has been made with the weak person	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request is pending and has not a calculated priority
Main post-condition(s)	
1	The filled checkbox is filled with at least one item and the coordinator has accept the calculated priority
Main Steps	
a	the actor `actCoordinator` executes the `oeReqCheckbox` use case
b	the actor `actCoordinator` executes the `oeSendFilledCheckbox` use case
c	the actor `actCoordinator` executes the `oeConfirmPriority` use case
Additional Information	
none	

Figure 2.6 shows the use case diagram for the ugAssignPriorityToHelpRequest usergoal use case

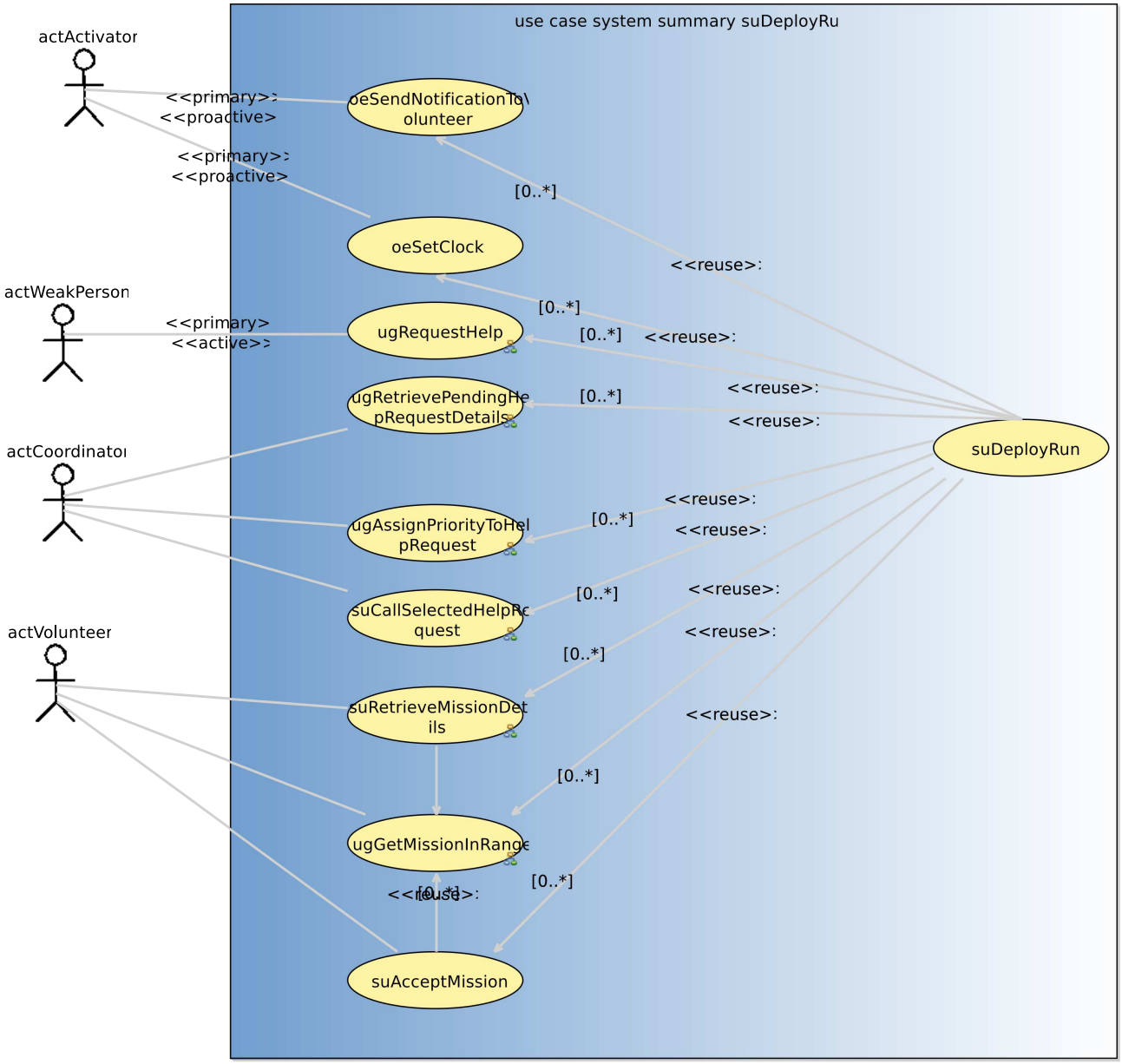


Figure 2.3: suDeployRun summary use case view

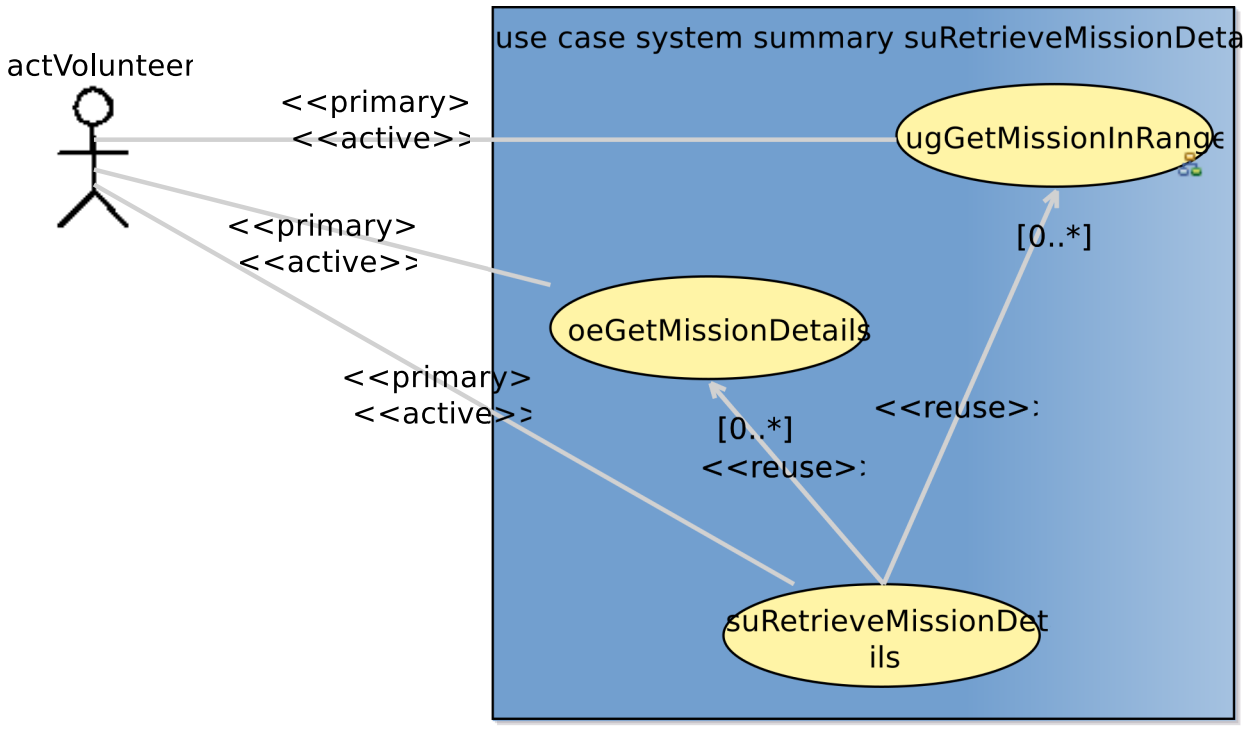


Figure 2.4: suRetrieveMissionDetails summary use case view

2.3.1.8 usergoal-ugGetCurrentPosition

This allows the Volunteer to save his current location in the system

Use-Case Description	
Name	ugGetCurrentPosition
Scope	system
Level	usergoal
Primary actor(s)	
1	`actVolunteer[active]`
Secondary actor(s)	
1	`actSensor[]`
Goal(s) description	
This allows the Volunteer to save his current location in the system	
Protocol condition(s)	
1	The system is running and ready
2	Volunteer is logged in
Pre-condition(s)	
1	The coordinates exists on the earth
Main post-condition(s)	
1	The coordinates are linked to this volunteer in saved for further use until new request of current position
Main Steps	
a	the actor `actVolunteer` executes the `oeGetCurrentPosition` use case

continues in next page . . .

... Use-Case Description table continuation

b	the actor `actSensor` executes the `oeSendCurrentPosition` use case
c	the actor `actVolunteer` executes the `oeRegisterPosition` use case
Additional Information	
none	

Figure 2.7 shows the use case diagram for the ugGetCurrentPosition usergoal use case

2.3.1.9 usergoal-ugGetMissionInRange

The actVolunteer's goal is to retrieve help requests that are in a specific range

Use-Case Description	
Name	ugGetMissionInRange
Scope	system
Level	usergoal
Primary actor(s)	
1	`actVolunteer[active]`
Secondary actor(s)	
1	`actPositionRequester[]`
2	`actSensor[]`
Goal(s) description	
The actVolunteer's goal is to retrieve help requests that are in a specific range	
Protocol condition(s)	
1	The system has been deployed.
Pre-condition(s)	
1	
Main post-condition(s)	
1	The system returns a non null list of HelpRequest or a message indicating that none has been found within the specified range
Main Steps	
a	the actor `actPositionRequester` executes the `oeGetPositionFromSensor` use case
b	the actor `actSensor` executes the `oeSendPos` use case
c	the actor `actVolunteer` executes the `oeGetInRangeMission` use case
Additional Information	
none	

Figure 2.8 shows the use case diagram for the ugGetMissionInRange usergoal use case

2.3.1.10 usergoal-ugRetrievePendingHelpRequestDetails

The goal is call a list with the help requests

Use-Case Description	
Name	ugRetrievePendingHelpRequestDetails
Scope	system

continues in next page . . .

... Use-Case Description table continuation

Level	usergoal
Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is call a list with the help requests	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	There is at least one pending help request
Main post-condition(s)	
1	The system sends a list with pending requests
Main Steps	
a	the actor `actCoordinator` executes the `oeGetPendingHelpRequests` use case
b	the actor `actCoordinator` executes the `oeGetHelpRequestDetail` use case
Additional Information	
none	

Figure 2.9 shows the use case diagram for the ugRetrievePendingHelpRequestDetails usergoal use case

2.3.1.11 subfunction-oeAcceptMission

The operation allow the Volunteer to accept a specific mission

USE-CASE DESCRIPTION	
Name	oeAcceptMission
Scope	system
Level	subfunction
Primary actor(s)	
1	`actVolunteer[active]`
Goal(s) description	
The operation allow the Volunteer to accept a specific mission	
Protocol condition(s)	
1	The volunteer has triggered suRetrieveMissionDetails for this specific mission
Pre-condition(s)	
1	
Main post-condition(s)	
1	The @post value of assigned HelpRequest ID of this volunteer instance is the selected HelpRequest ID
Additional Information	
none	

2.3.1.12 subfunction-oeConfirmPriority

The goal is to confirm the calculated priority

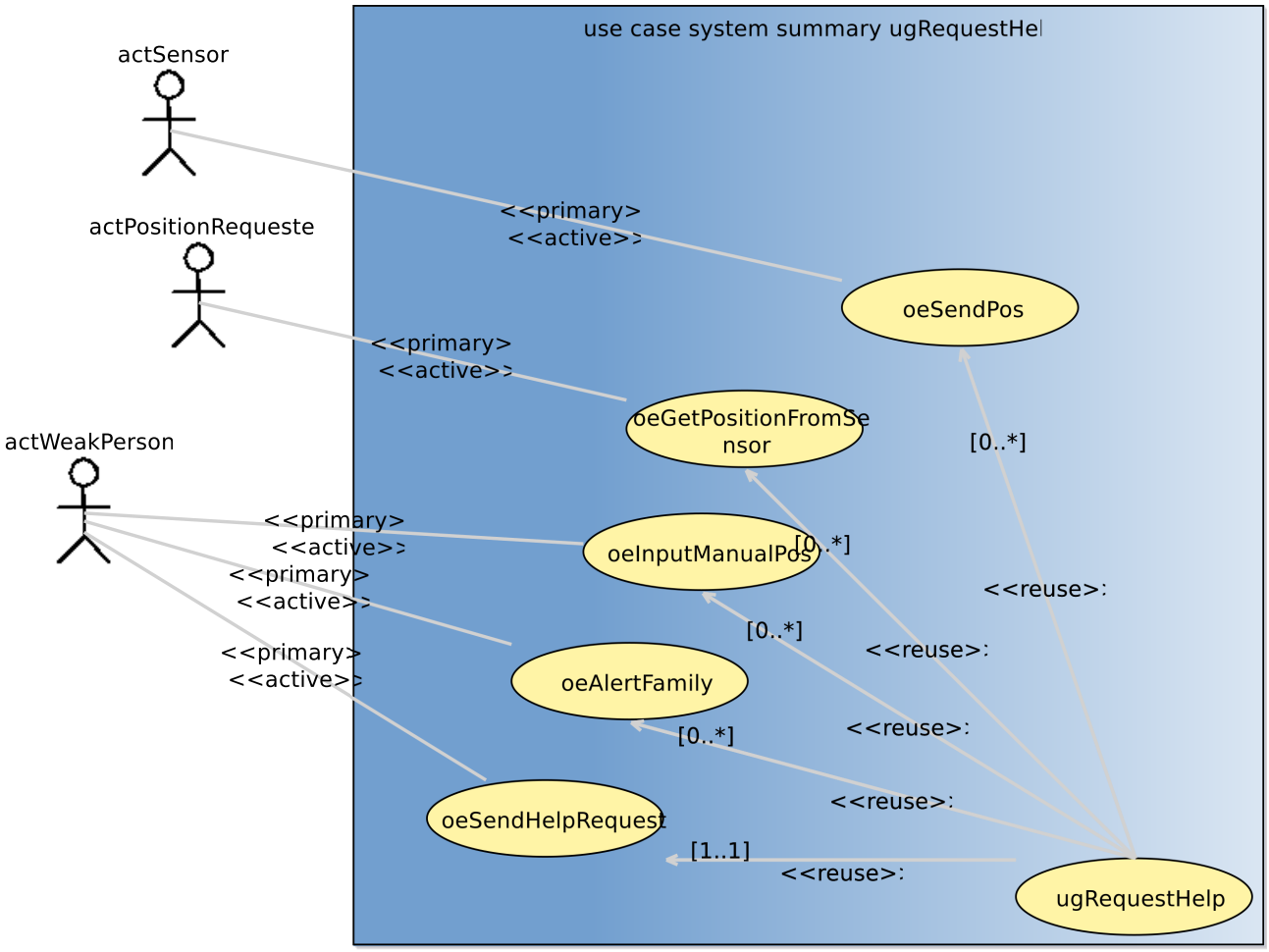


Figure 2.5: ugReauestHelp usergoal use case view

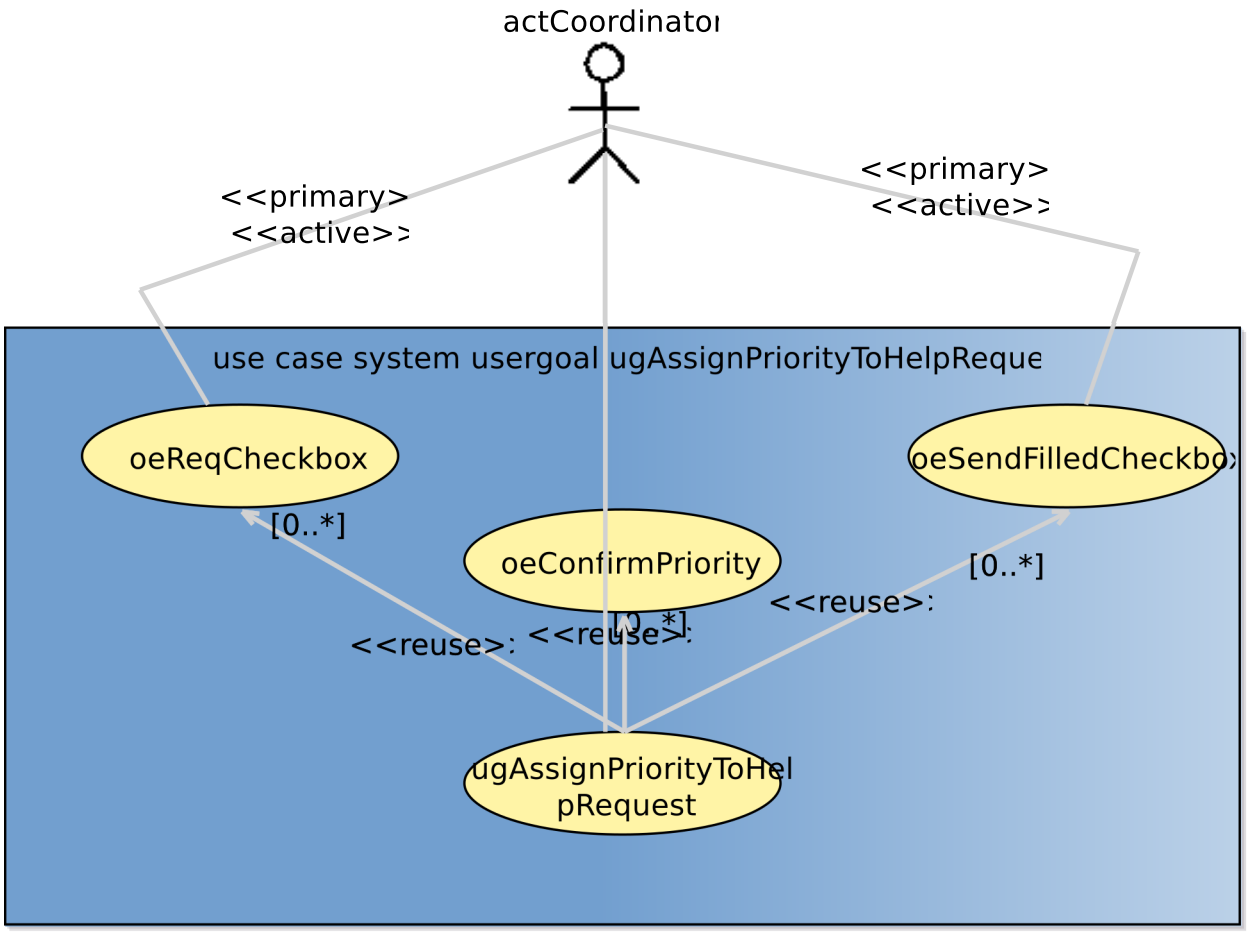


Figure 2.6: ugAssignPriorityToHelpRequest usergoal use case view

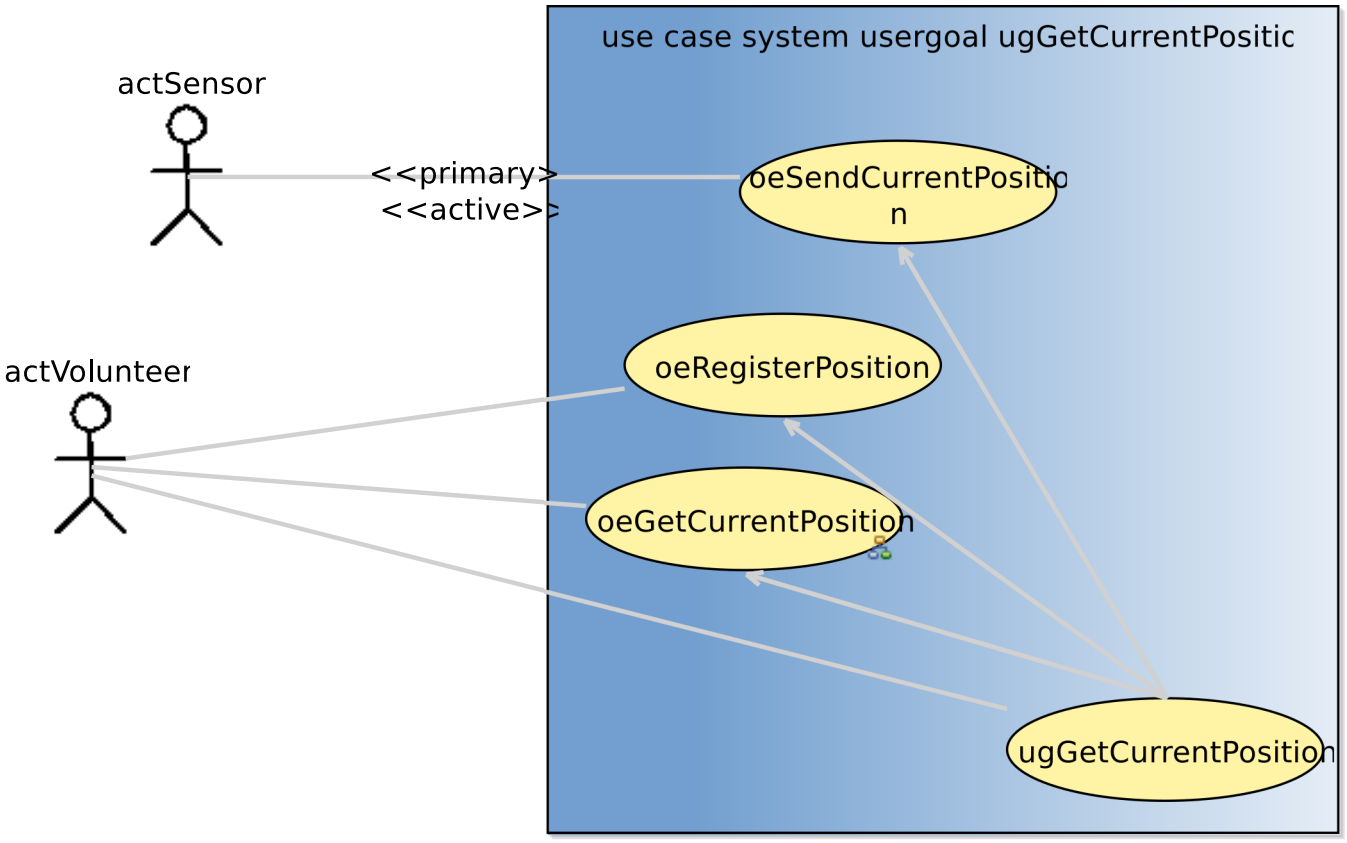


Figure 2.7: ugGetCurrentPosition usergoal use case view

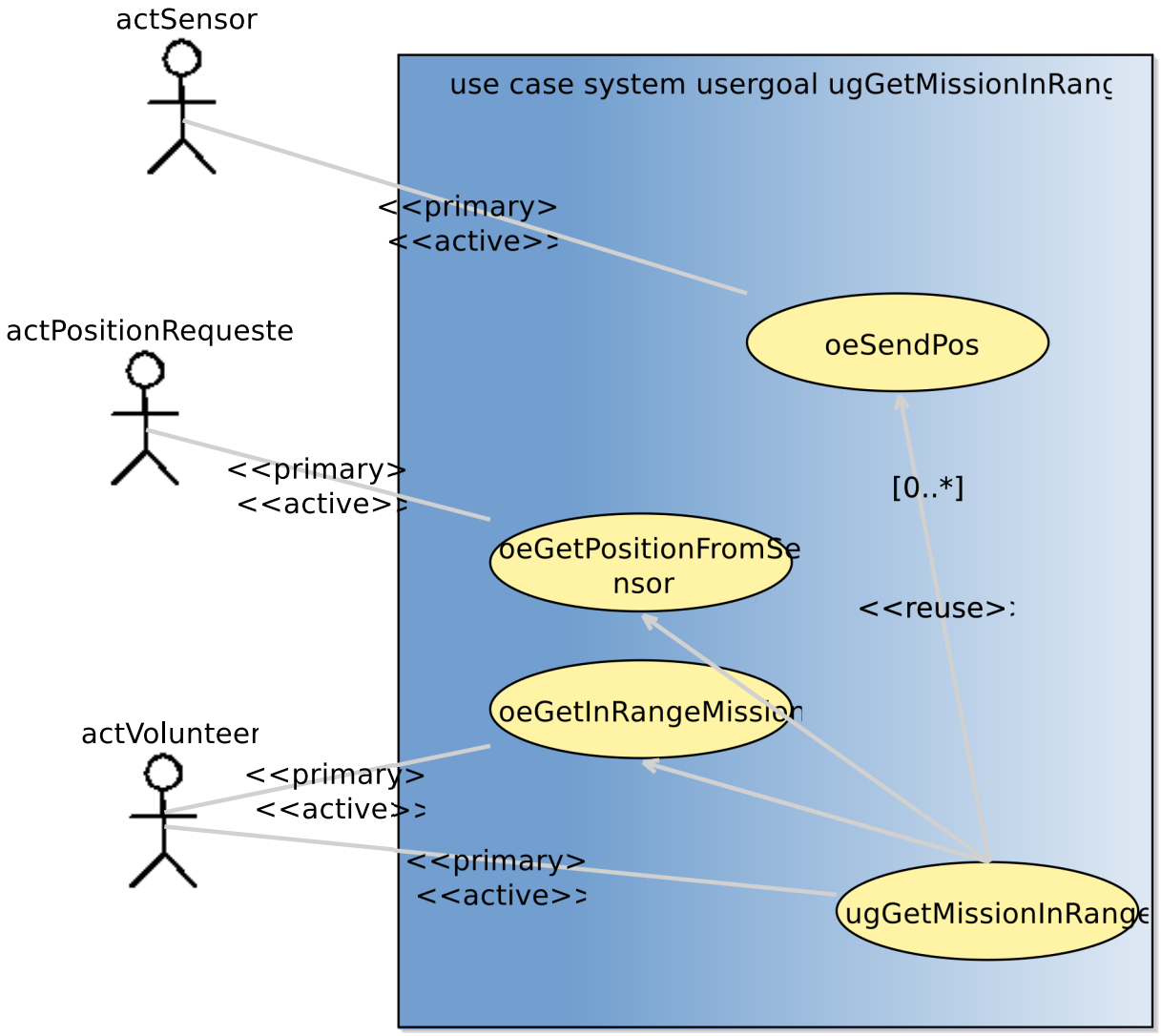


Figure 2.8: ugGetMissionInRange usergoal use case view

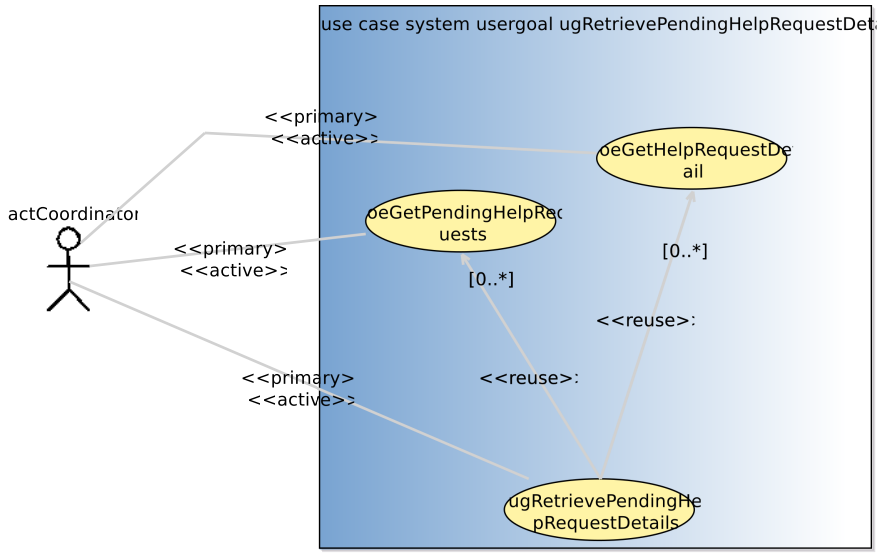


Figure 2.9: ugRetrievePendingHelpRequestDetails usergoal use case view

Use-Case Description	
Name	oeConfirmPriority
Scope	system
Level	subfunction
Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is to confirm the calculated priority	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request is on the pending list and the priority has been successfully calculated
Main post-condition(s)	
1	The system assigns the priority to the help request
Additional Information	
none	

2.3.1.13 subfunction-oeGetConfirm

The goal is to get the confirmation from the phone company

Use-Case Description	
Name	oeGetConfirm
Scope	system
Level	subfunction
Primary actor(s)	
1	`actPhoneCompany[active]`
Goal(s) description	
The goal is to get the confirmation from the phone company	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request exists and the coordinator is logged in
Main post-condition(s)	
1	The phone company sends a confirmation that the number exists and the phone is reachable

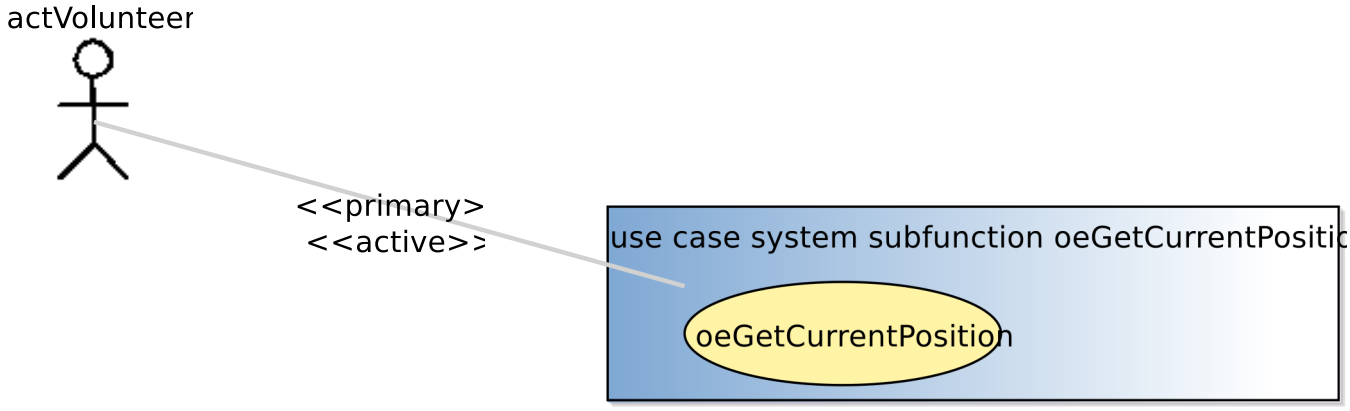


Figure 2.10: oeGetCurrentPosition subfunction use case view

Use-Case Description	
Name	oeGetPendingHelpRequests
Scope	system
Level	subfunction
Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is to get a list with the pending help requests	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	There is at least one pending help request
Main post-condition(s)	
1	
Additional Information	
none	

2.3.1.16 subfunction-oeReqCall

The goal is to request a call to the phone company

Use-Case Description	
Name	oeReqCall
Scope	system
Level	subfunction
Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is to request a call to the phone company	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request is on the pending list and connection between the system and the phone company is maintained
Main post-condition(s)	
1	The number of the requested call exists and the coordinator is connected to the weak person
Additional Information	
none	

... Use-Case Description table continuation

Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is to request a checkbox from the system	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request is still on the pending list and the coordinator has successfully called the weak person
Main post-condition(s)	
1	
Additional Information	
none	

2.3.1.18 subfunction-oeSendFilledCheckbox

The goal is to send a filled checkbox to the system

USE-CASE DESCRIPTION	
Name	oeSendFilledCheckbox
Scope	system
Level	subfunction
Primary actor(s)	
1	`actCoordinator[active]`
Goal(s) description	
The goal is to send a filled checkbox to the system	
Protocol condition(s)	
1	The system is running
Pre-condition(s)	
1	The help request is on the pending list and the checkbox is filled with at least one item
Main post-condition(s)	
1	The system is able to calculate the risk level of the weak person
Additional Information	
none	

2.3.2 Use Case Instance(s)

2.3.2.1 Use-Case Instance - uciAlertTheFamily:suAltertAFamilyMember

Figure 2.11 Use case instance describing Alert familly use case

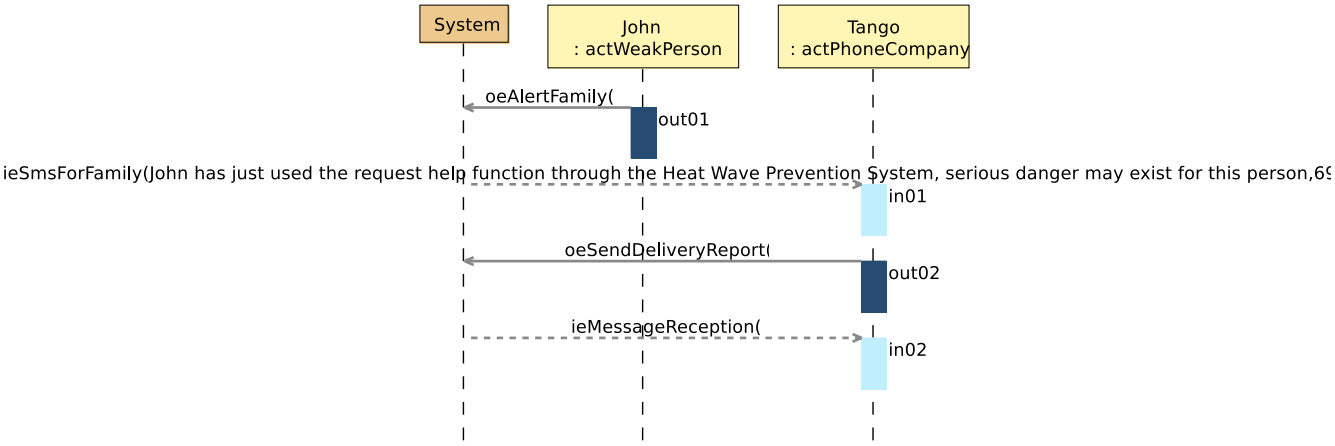


Figure 2.11:

2.3.2.2 Use-Case Instance - uciCallSelectedHelp:suCallSelectedHelpRequest

Call the weak person by selecting a help request it sent

SUMMARY USE-CASE INSTANCE
Instantiated Use Case suCallSelectedHelpRequest
Instance ID uciCallSelectedHelp

Figure 2.12 Use case instance describing use case Call selected help request

2.3.2.3 Use-Case Instance - uciNormalRun:suDeployRun

Use case instance for the summary use case suDeployRun illustrating a simple and complete between all actors

SUMMARY USE-CASE INSTANCE
Instantiated Use Case suDeployRun
Instance ID uciNormalRun

Figure 2.13 Use case instance describing a normal run of the application

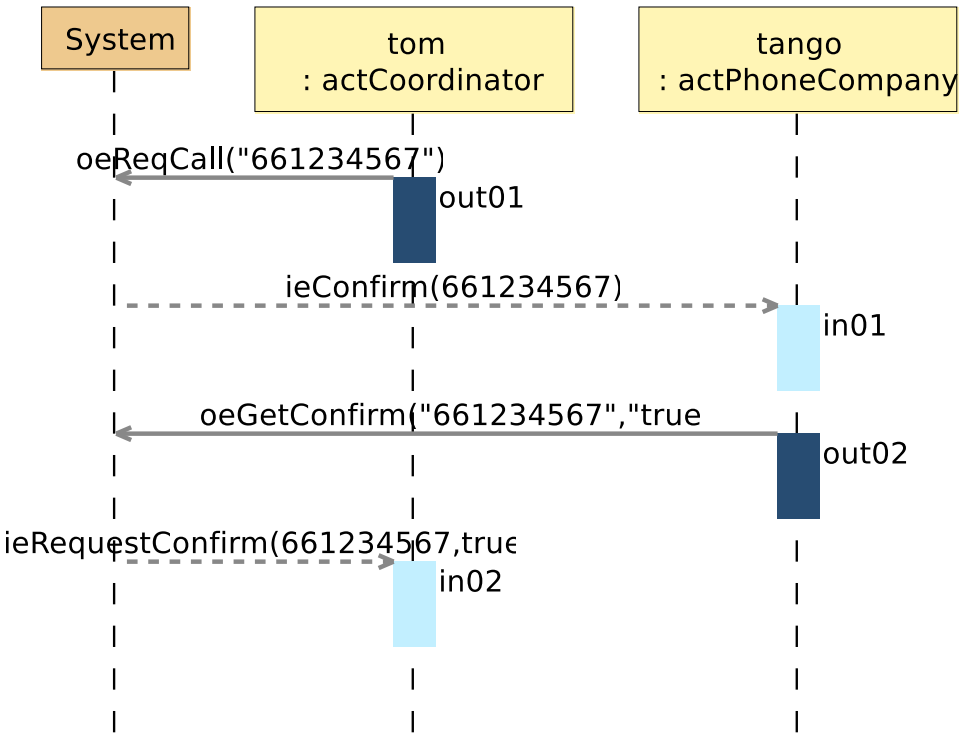


Figure 2.12:

2.3.2.4 Use-Case Instance - uciSendHelpRequest:ugRequestHelp

this is an instance that represent a weakPerson willing to request help

SUMMARY USE-CASE INSTANCE
Instantiated Use Case ugRequestHelp
Instance ID uciSendHelpRequest

Figure 2.14 Use case instance describing the help request summary of a weak person

2.3.2.5 Use-Case Instance - uciAssignPriority:ugAssignPriorityToHelpRequest

Assign Priority to the help request

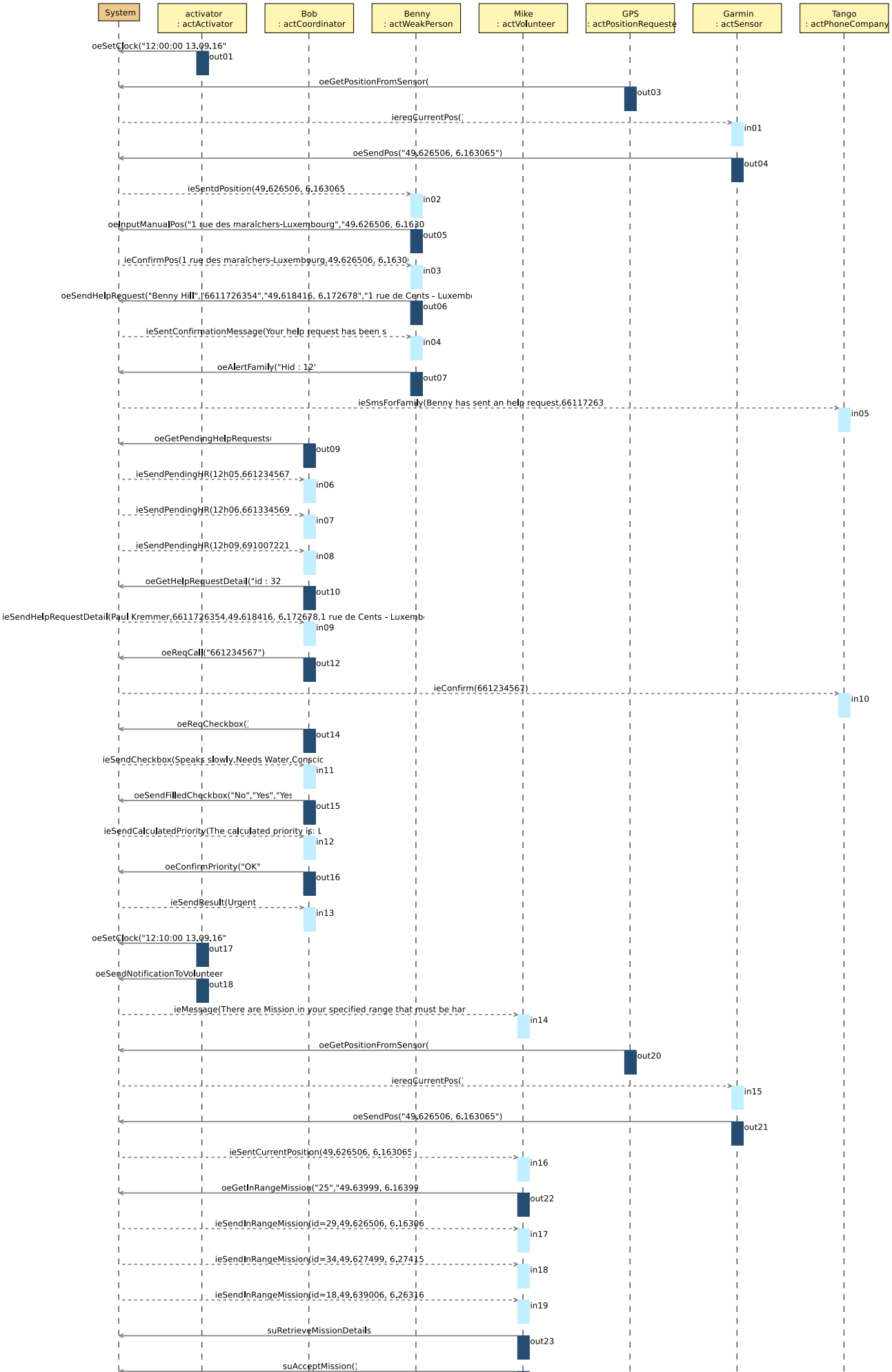
USERGOAL USE-CASE INSTANCE
Instantiated Use Case ugAssignPriorityToHelpRequest
Instance ID uciAssignPriority

Figure 2.15 Use case instance discribing Assign priority user goal

2.3.2.6 Use-Case Instance - uciGetCurrentPositon:ugGetCurrentPosition

This represent a use case instance of Volunteer willing to register his position into the system

USERGOAL USE-CASE INSTANCE


System
activator : actActivator
Bob : actCoordinator
Benny : actWeakPerson
Mike : actVolunteer
GPS : actPositionRequeste
Garmin : actSensor
Tango : actPhoneCompany
oeSetClock("12:00:00 13.09.16"
out01
oeGetPositionFromSensor(
out03
iereqCurrentPos(
in01
oeSendPos("49.626506, 6.163065")
out04
ieSentdPosition(49.626506, 6.163065
in02
oeInputManualPos("1 rue des maraîchers-Luxembourg","49.626506, 6.1630
out05
ieConfirmPos(1 rue des maraîchers-Luxembourg,49.626506, 6.1630
in03
oeSendHelpRequest("Benny Hill","6611726354","49.618416, 6.172678","1 rue de Cents - Luxembo
out06
ieSentConfirmationMessage(Your help request has been s
in04
oeAlertFamily("Hid : 12'
out07
ieSmsForFamily(Benny has sent an help request,66117263
in05
oeGetPendingHelpRequests
out09
ieSendPendingHR(12h05,661234567
in06
ieSendPendingHR(12h06,661334569
in07
ieSendPendingHR(12h09,691007221
in08
oeGetHelpRequestDetail("id : 32
out10
ieSendHelpRequestDetail(Paul Kremmer,6611726354,49.618416, 6.172678,1 rue de Cents - Luxembo
in09
oeReqCall("661234567")
out12
ieConfirm(661234567)
in10
oeReqCheckbox(
out14
ieSendCheckbox(Speaks slowly,Needs Water,Conscic
in11
oeSendFilledCheckbox("No","Yes","Yes
out15
ieSendCalculatedPriority(The calculated priority is: L
in12
oeConfirmPriority("OK"
out16
ieSendResult(Urgent
in13
oeSetClock("12:10:00 13.09.16"
out17
oeSendNotificationToVolunteer
out18
ieMessage(There are Mission in your specified range that must be har
in14
oeGetPositionFromSensor(
out20
iereqCurrentPos(
in15
oeSendPos("49.626506, 6.163065")
out21
ieSentCurrentPosition(49.626506, 6.163065
in16
oeGetInRangeMission("25","49.63999, 6.16399
out22
ieSendInRangeMission(id=29,49.626506, 6.16306
in17
ieSendInRangeMission(id=34,49.627499, 6.27415
in18
ieSendInRangeMission(id=18,49.639006, 6.26316
in19
suRetrieveMissionDetails
out23
suAcceptMission()

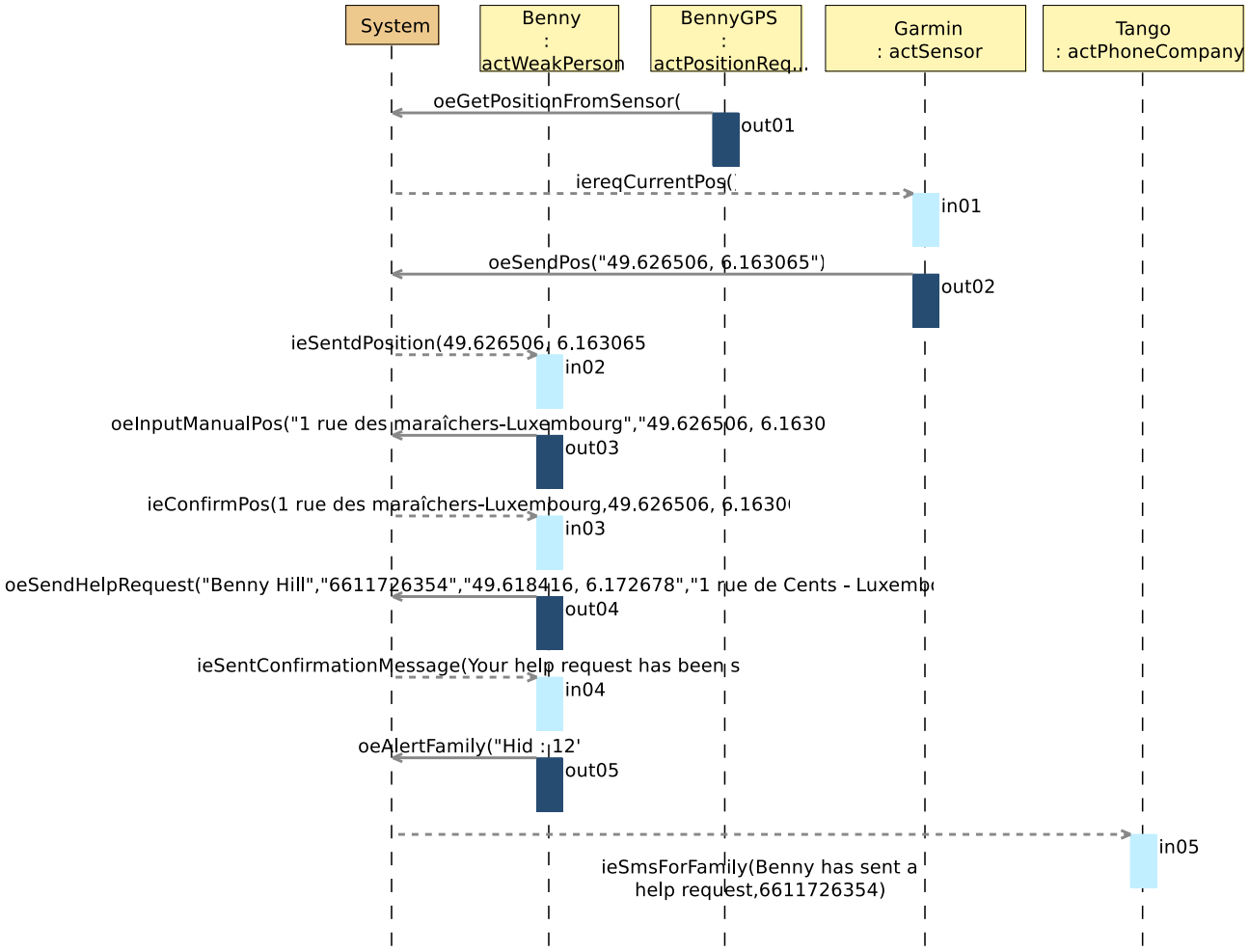

System
Benny : actWeakPerson
BennyGPS : actPositionReq...
Garmin : actSensor
Tango : actPhoneCompany
oeGetPositionFromSensor(
out01
iereqCurrentPos(
in01
oeSendPos("49.626506, 6.163065")
out02
ieSentdPosition(49.626506, 6.163065
in02
oeInputManualPos("1 rue des maraîchers-Luxembourg","49.626506, 6.1630
out03
ieConfirmPos(1 rue des maraîchers-Luxembourg,49.626506, 6.1630
in03
oeSendHelpRequest("Benny Hill","6611726354","49.618416, 6.172678","1 rue de Cents - Luxemb
out04
ieSentConfirmationMessage(Your help request has been s
in04
oeAlertFamily("Hid : 12'
out05
in05
ieSmsForFamily(Benny has sent a help request,6611726354)

Figure 2.14:

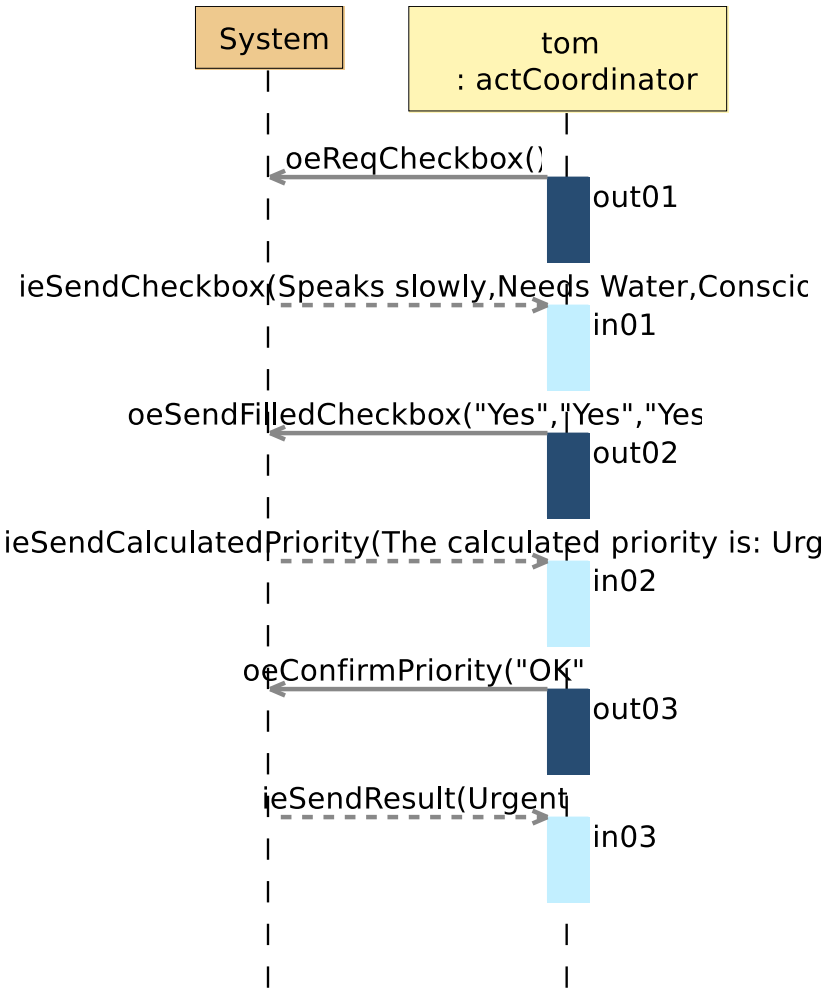

System
tom
: actCoordinator
oeReqCheckbox()
out01
ieSendCheckbox(Speaks slowly,Needs Water,Conscic
in01
oeSendFilledCheckbox("Yes","Yes","Yes
out02
ieSendCalculatedPriority(The calculated priority is: Urg
in02
oeConfirmPriority("OK"
out03
ieSendResult(Urgent
in03

Figure 2.15:

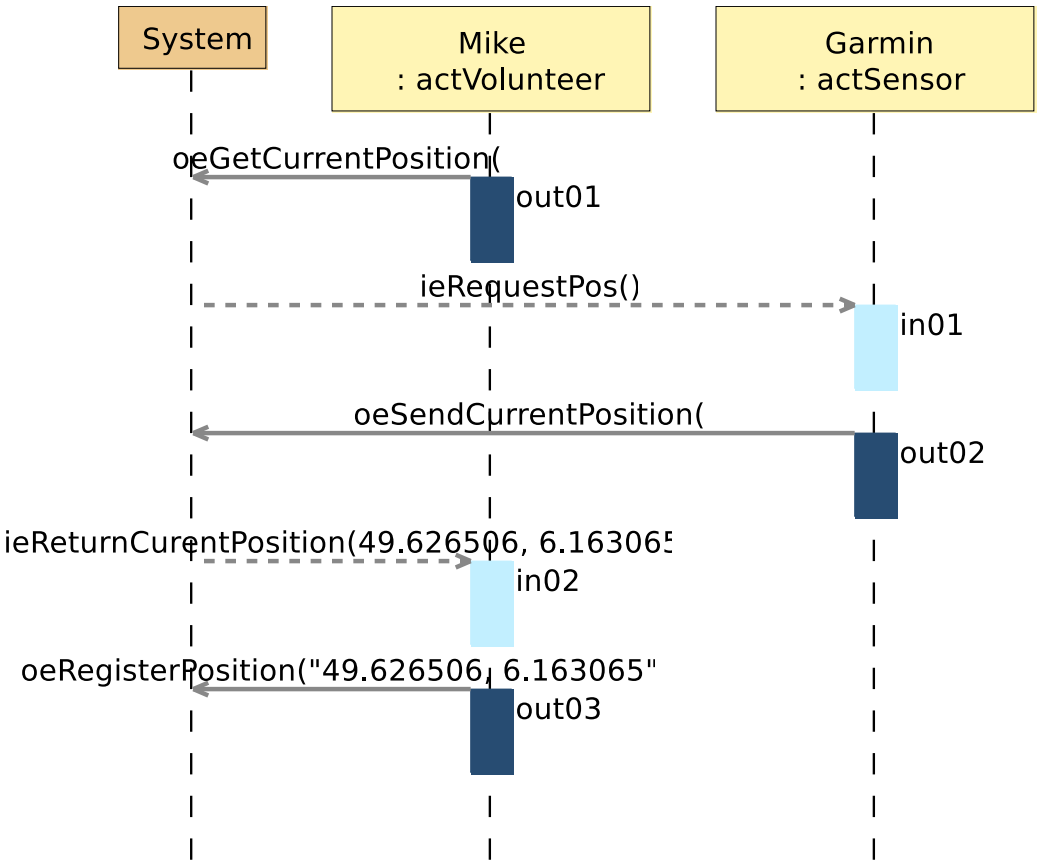

System
Mike
: actVolunteer
Garmin
: actSensor
oeGetCurrentPosition(
out01
ieRequestPos()
in01
oeSendCurrentPosition(
out02
ieReturnCurentPosition(49.626506, 6.16306
in02
oeRegisterPosition("49.626506, 6.163065"
out03

Figure 2.16:

Figure 2.17 Use case instance describing The get in range mission of a Volunteer

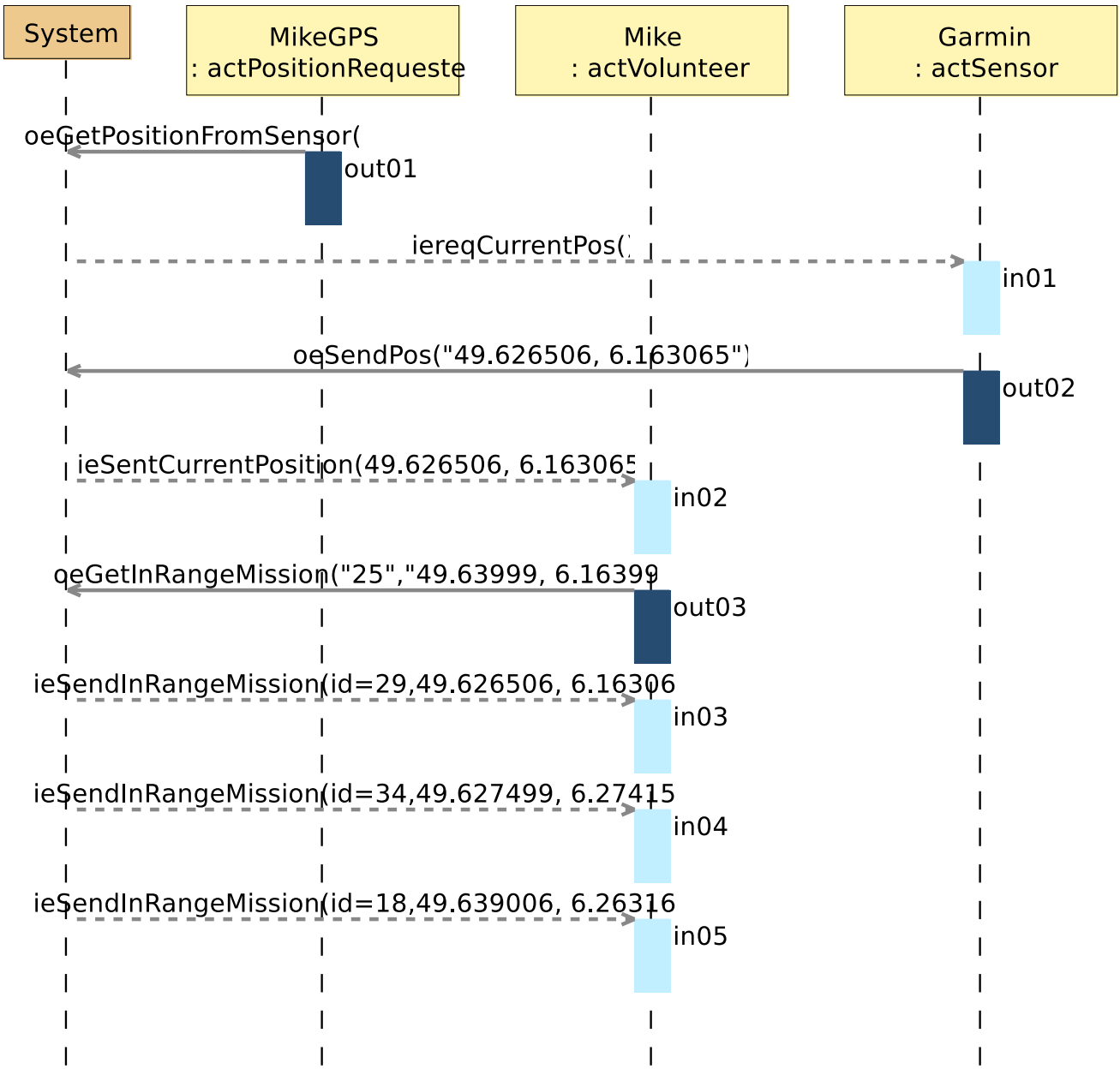


Figure 2.17:

2.3.2.8 Use-Case Instance - uciGetPendingHelpRequests:ugRetrievePendingHelpRequestDetails

Receive a list with all the pending help requests.

USERGOAL USE-CASE INSTANCE
Instantiated Use Case ugRetrievePendingHelpRequestDetails

continues in next page . . .

. . . usergoal Use-Case Instance table continuation

Instance ID
uciGetPendingHelpRequests

Figure 2.18 Use case instance describing the get pending help requests

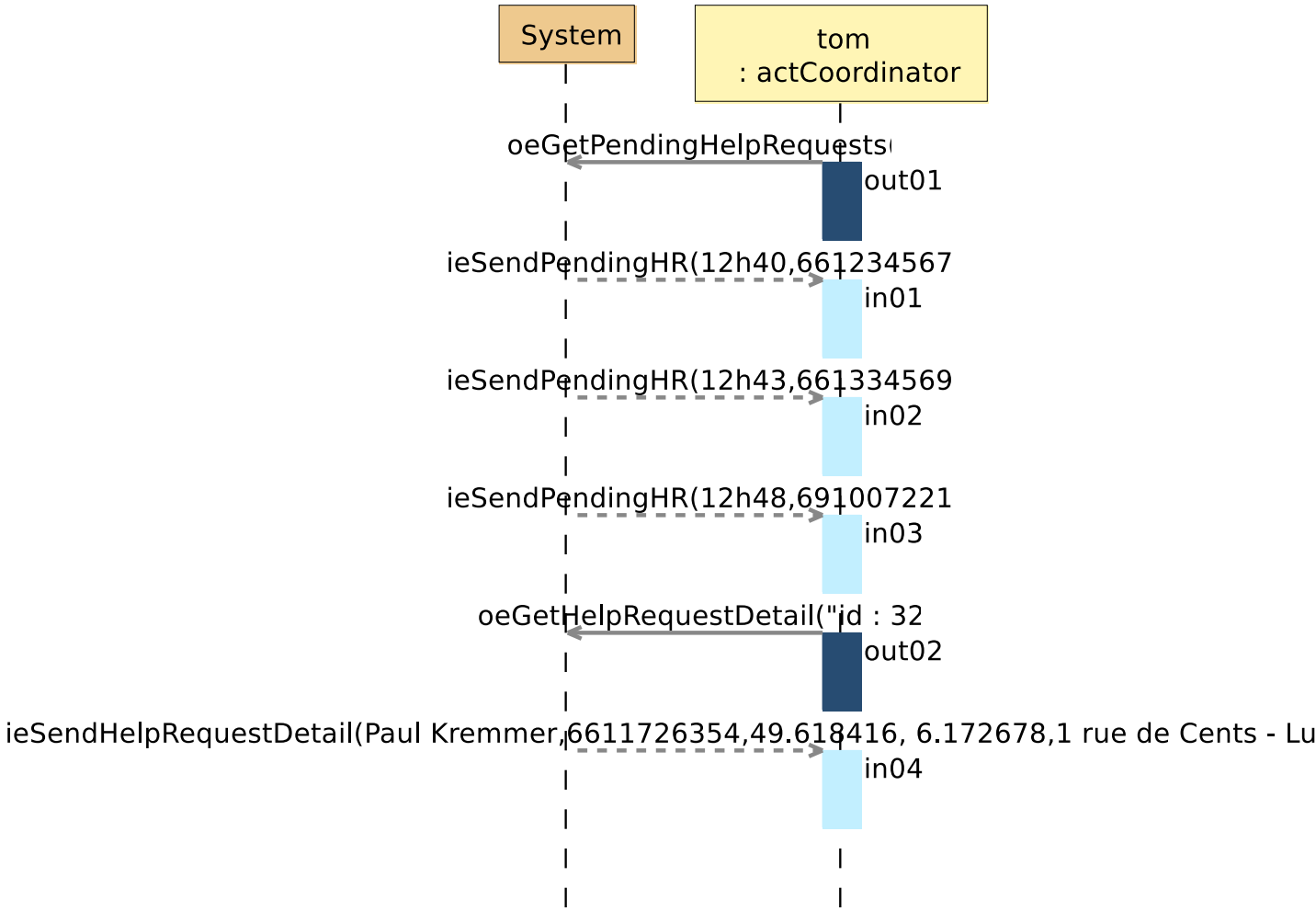

System
tom
: actCoordinator
oeGetPendingHelpRequests(
out01
ieSendPendingHR(12h40,661234567
in01
ieSendPendingHR(12h43,661334569
in02
ieSendPendingHR(12h48,691007221
in03
oeGetHelpRequestDetail("id : 32
out02
ieSendHelpRequestDetail(Paul Kremmer,6611726354,49.618416, 6.172678,1 rue de Cents - Lu
in04

Figure 2.18:

Chapter 3

Environment Model

We provide below the view(s) defined for the Messir environment model (cf. [1]) of the system.

3.1 Local view 06

Figure 3.1 Shows the local view for the activator actor and interface

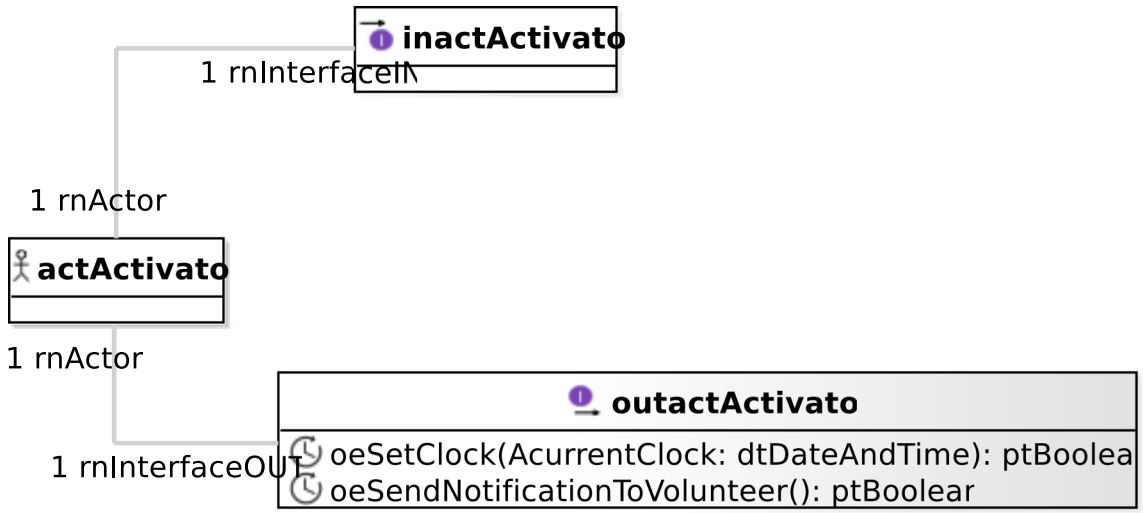


Figure 3.1: Environment Model - Local View 06. Environment model local view.

3.2 Local view 09

Figure 3.2 shows the local view for the coordinator actor and interface

3.3 Local view 10

Figure 3.3 shows Environment local view from volunteer actor and interface

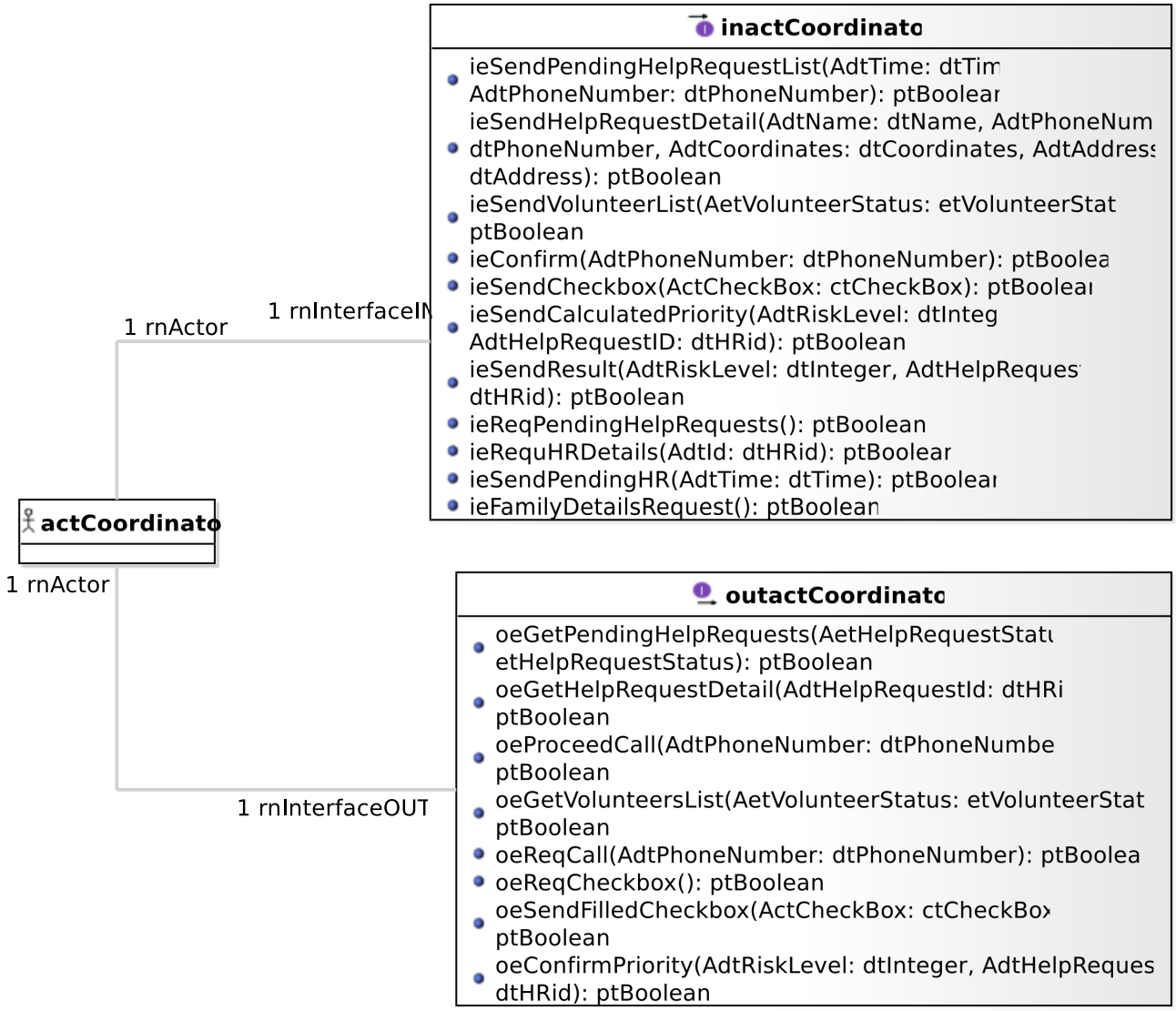


Figure 3.2: Environment Model - Local View 09. Environment model local view.

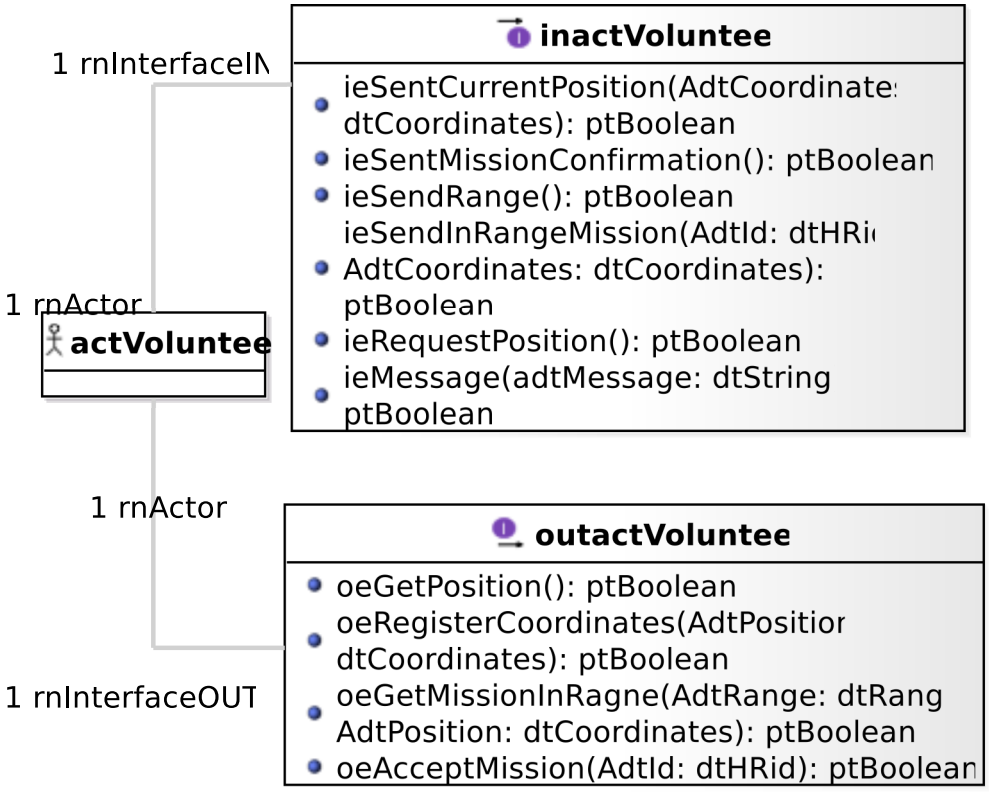


Figure 3.3: Environment Model - Local View 10. Environment local view.

3.4 Local view 11

Figure 3.4 shows environment viwe from Weak person actor and interface

3.5 Actors and Interfaces Descriptions

We provide for the given views the description of the actors together with their associated input and output interface descriptions.

3.5.1 `actActivator` Actor

Actor	
actActivator	
represents a logical actor for time automatic message sending based on system's or environment status.	
OutputInterfaces	
OUT 1	**`[proactive] oeSetClock(AcurrentClock:dtDateAndTime):ptBoolean`**
	Used to update clock global variable
OUT 2	**`[proactive] oeSendNotificationToVolunteer():ptBoolean`**
	Used to trigger a message to volunteer if a low priority crisis is not handled for too long and is in range of certain volunteer

3.5.2 `actAuthPerson` Actor

Actor
actAuthPerson
used to give access to other actors by giving the information as input

... Actor table continuation

OUT 3	**oeProceedCall(AdtPhoneNumber:dtPhoneNumber):ptBoolean** send to call a specific ctWeakPerson instance.
OUT 4	**oeGetVolunteersList(AetVolunteerStatus:etVolunteerStatus):ptBoolean** send to request all ctVolunteers instances with a certain status.
OUT 5	**oeReqCall(AdtPhoneNumber:dtPhoneNumber):ptBoolean** send to request a call to a specific ctWeakPerson instance from a ctPhoneCompany.
OUT 6	**oeReqCheckbox():ptBoolean**
OUT 7	**oeSendFilledCheckbox(ActCheckBox:ctCheckBox):ptBoolean**
OUT 8	**oeConfirmPriority(AdtRiskLevel:dtInteger, AdtHelpRequestID:dtHRid):ptBo** send to attach a risk level to a specific ctWeakPerson instance.
InputInterfaces	
IN 1	**ieSendPendingHelpRequestList(AdtTime:dtTime, AdtPhoneNumber:dtPhoneNumber):ptBoolean** its reception lists all ctHelpRequest instances having a specific status.
IN 2	**ieSendHelpRequestDetail(AdtName:dtName, AdtPhoneNumber:dtPhoneNumber, AdtCoordinates:dtCoordinates, AdtAddress:dtAddress):ptBoolean** allows to receive requested details of a specific ctHelpRequest instance.
IN 3	**ieSendVolunteerList(AetVolunteerStatus:etVolunteerStatus):ptBoolean** its reception lists all ctVolunteer instances having a specific status.
IN 4	**ieConfirm(AdtPhoneNumber:dtPhoneNumber):ptBoolean** its reception indicates if the requested number exists.
IN 5	**ieSendCheckbox(ActCheckBox:ctCheckBox):ptBoolean**
IN 6	**ieSendCalculatedPriority(AdtRiskLevel:dtInteger, AdtHelpRequestID:dtHRid):ptBoolean** allows to receive a calculated risk level.
IN 7	**ieSendResult(AdtRiskLevel:dtInteger, AdtHelpRequestID:dtHRid):ptBoolean** allows to receive a ctHelpRequest instance with a calculated risk level.
IN 8	**ieReqPendingHelpRequests():ptBoolean**
IN 9	**ieRequHRDetails(AdtId:dtHRid):ptBoolean**
IN 10	**ieSendPendingHR(AdtTime:dtTime):ptBoolean** allows to receive a specific ctHelpRequest instance.
IN 11	**ieFamilyDetailsRequest():ptBoolean** allows to receive details of family from a specific ctWeakPerson instance.

3.5.4 `actMsrCreator` Actor

ACTOR	
actMsrCreator actor used to create the system and the environment	
OutputInterfaces	
OUT 1	**oeCreateSystemAndEnvironment():ptBoolean**

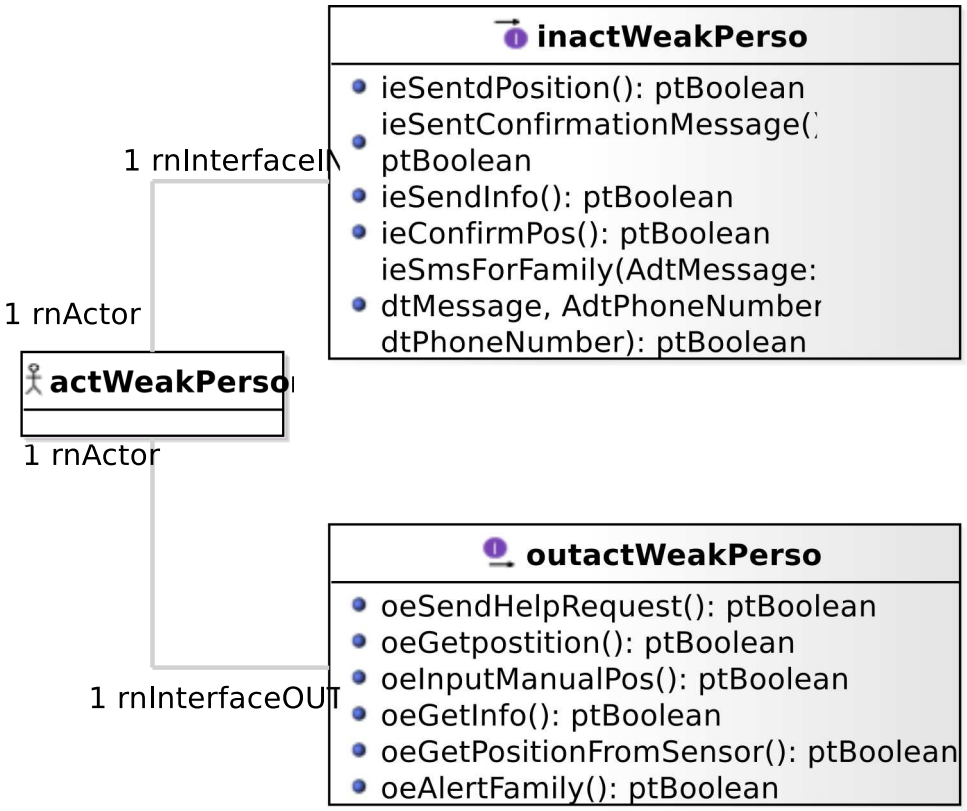


Figure 3.4: Environment Model - Local View 11. Environment local view.

3.5.5 `actPhoneCompany` Actor

ACTOR	
actPhoneCompany Env Phone company	
OutputInterfaces	
OUT 1	`oeGetConfirm():ptBoolean`
OUT 2	`oeSendDeliveryReport(AdtMessage:dtMessage, AdtPhoneNumber:dtPhoneNumber`
InputInterfaces	
IN 1	`ieRequestConfirm(AdtPhoneNumber:dtPhoneNumber):ptBoolean`
IN 2	`ieSmsForFamily(AdtMessage:dtMessage, AdtPhoneNumber:dtPhoneNumber):ptBo`

3.5.6 `actPositionInputActor` Actor

ACTOR	
actPositionInputActor Env PositionInputActor	
OutputInterfaces	
OUT 1	`oeInputPost():ptBoolean`
InputInterfaces	
IN 1	`ieSentPosition():ptBoolean`

3.5.7 `actPositionRequester` Actor

ACTOR	
actPositionRequester Env PositionRequester	
OutputInterfaces	
OUT 1	`oeGetPositionFromSensor():ptBoolean`
InputInterfaces	
IN 1	`ieSendSensorPosition():ptBoolean`

3.5.8 `actSensor` Actor

ACTOR
actSensor Env Sensor

continues in next page . . .

. . . Actor table continuation

OutputInterfaces	
OUT 1	**oeSendPos():ptBoolean** sent to update instance of ctHuman coordinates attributes
InputInterfaces	
IN 1	**iereqCurrentPos():ptBoolean** sent to trigger the oeSentCurrent pos from actSensor
IN 2	**ieReturnCurentPosition(adtCoordinates:dtCoordinates):ptBoolean** sent to trigger the oeSentCurrent pos from actSensor

3.5.9 `actVolunteer` Actor

ACTOR	
actVolunteer Env Volunteer	
OutputInterfaces	
OUT 1	**oeGetPosition():ptBoolean**
OUT 2	**oeGetMissionInRagne(AdtRange:dtRange, AdtPosition:dtCoordinates):ptBoolean** Sent to retrieve instances of ctHelpRequest
OUT 3	**oeAcceptMission(AdtId:dtHRid):ptBoolean** Sent to confirm handling of instance of ctHelpRequest
OUT 4	**oeRegisterCoordinates(AdtPosition:dtCoordinates):ptBoolean** Sent to save dtCoordinates into the instance of ctVolunteer
InputInterfaces	
IN 1	**ieSentCurrentPosition(AdtCoordinates:dtCoordinates):ptBoolean** Its reception confirms dtCorrdinates
IN 2	**ieSentMissionConfirmation():ptBoolean** Its reception confirms the handling of the HelpRequest
IN 3	**ieSendRange():ptBoolean** Its reception will trigger the sensor to send current coordinates
IN 4	**ieSendInRangeMission(AdtId:dtHRid, AdtCoordinates:dtCoordinates):ptBoolean** Allows get instances of ceHelpRequest which are in dtRange
IN 5	**ieRequestPosition():ptBoolean**

3.5.10 `actWeakPerson` Actor

ACTOR	
actWeakPerson Env WeakPerson	
OutputInterfaces	
OUT 1	**oeSendHelpRequest():ptBoolean** Sent to create a ctHelpRequest instance from the corresponding ctWeakPerson instance
OUT 2	**oeGetpostition():ptBoolean**
OUT 3	**oeInputManualPos():ptBoolean**

continues in next page . . .

... Actor table continuation

	sent to request confirmation of the manually entered dtCoordinates
OUT 4	**oeGetInfo():ptBoolean**
OUT 5	**oeGetPositionFromSensor():ptBoolean** sent to request the dtCoordinates of the ctWeakPerson instance from the ctSensor instance
OUT 6	**oeAlertFamily():ptBoolean** sent to request to send a SMS to all ctWeakPersonFamily instances of the corresponding ctWeakPerson instance from ctPhoneCompany instance
InputInterfaces	
IN 1	**ieSentdPosition():ptBoolean** Its reception are the dtCoordinates of the corresponding ctWeakPerson instance
IN 2	**ieSentConfirmationMessage():ptBoolean** Its reception is a confirmation message that the created ctHelpRequest instance was received by the system
IN 3	**ieSendInfo():ptBoolean**
IN 4	**ieConfirmPos():ptBoolean** Its reception confirms the dtCoordinates entered manually
IN 5	**ieSmsForFamily(AdtMessage:dtMessage, AdtPhoneNumber:dtPhoneNumber):ptBo** Its reception confirms the SMS was received by ctPhoneCompany instance

3.5.11 `actWeakPersonFamily` Actor

ACTOR	
actWeakPersonFamily Env WeakPersonFamily	
OutputInterfaces	
OUT 1	**oeSubscribe():ptBoolean**
OUT 2	**oeConfirmMessage():ptBoolean**
OUT 3	**oeConfirmCall():ptBoolean**
InputInterfaces	
IN 1	**ieSentPosition():ptBoolean**
IN 2	**ieGetMessage():ptBoolean**
IN 3	**ieGetCall():ptBoolean**

Chapter 4

Concept Model

4.1 PrimaryTypes-Classes

4.1.1 Local view 02

Figure 4.1 Gloabal view of all class types relations

4.1.2 Local view 03

Figure 4.2 Relation view representation aggregation between Help request and pending help requests

4.1.3 Local view 04

Figure 4.3 Relation diagram representing relation between Weak person and help request

4.1.4 Local view 05

Figure 4.4 Relation view showing Relation between Human and Coordinator, weak person an Volunteer

4.1.5 Local view 07

Figure 4.5 Shows the environment model global view

4.2 Concept Model Types Descriptions

This section provides the textual descriptions of all the types defined in the concept model and that can be part of the graphical views provided.

4.2.1 Primary types - Class types descriptions

The table below is providing comments on the graphical views given for the class types of the primary types. Type logical operations are precisely specified in the operation model.

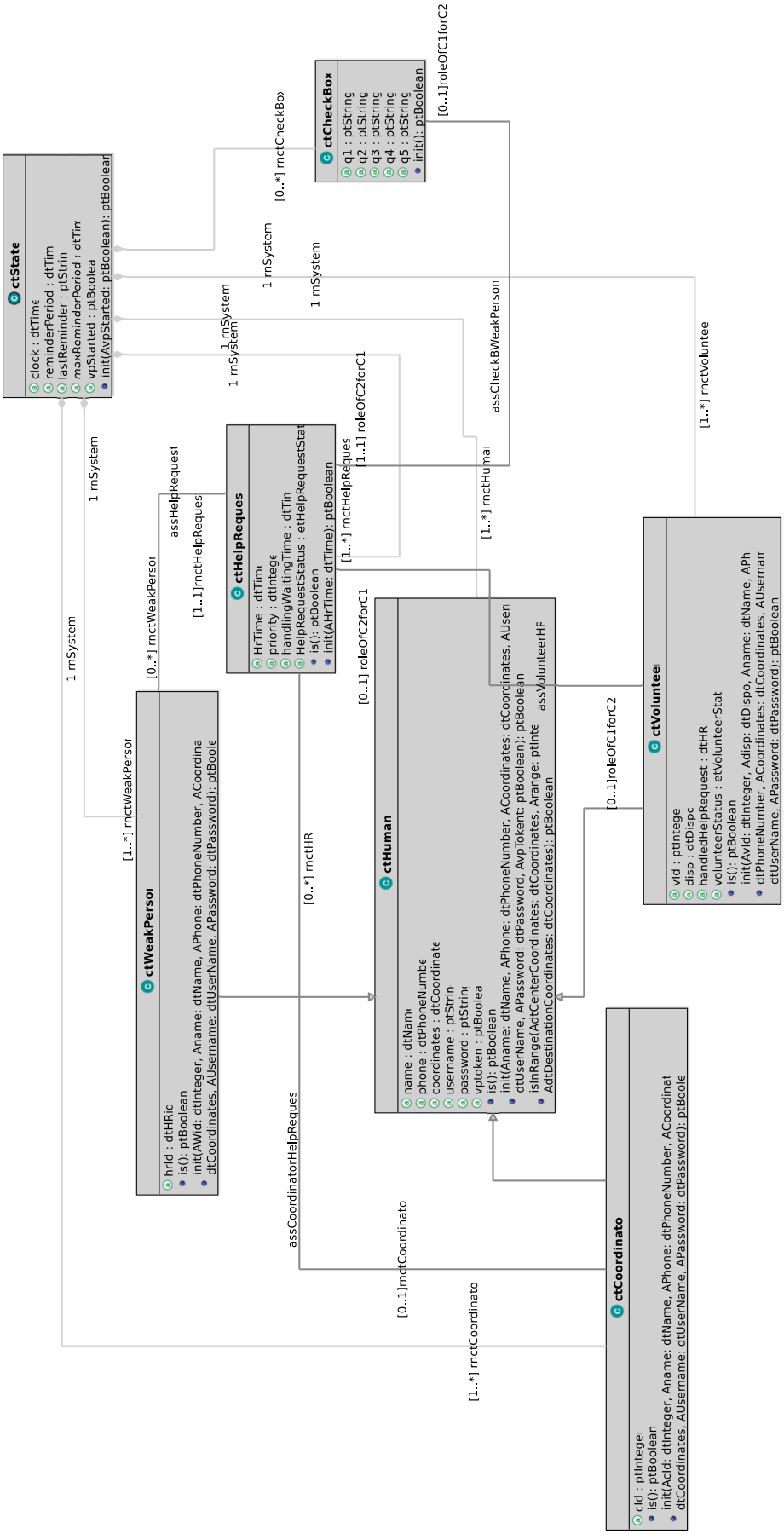


Figure 4.1: Concept Model - PrimaryTypes-Classes local view 02. shows concept model global view.

Figure 4.2: Concept Model - PrimaryTypes-Classes local view 03. Concept model local view.

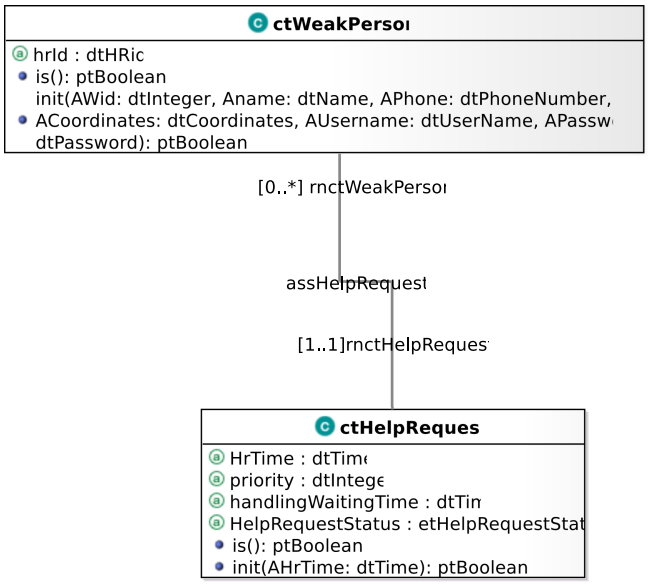


Figure 4.3: Concept Model - PrimaryTypes-Classes local view 04. Concept model local view.

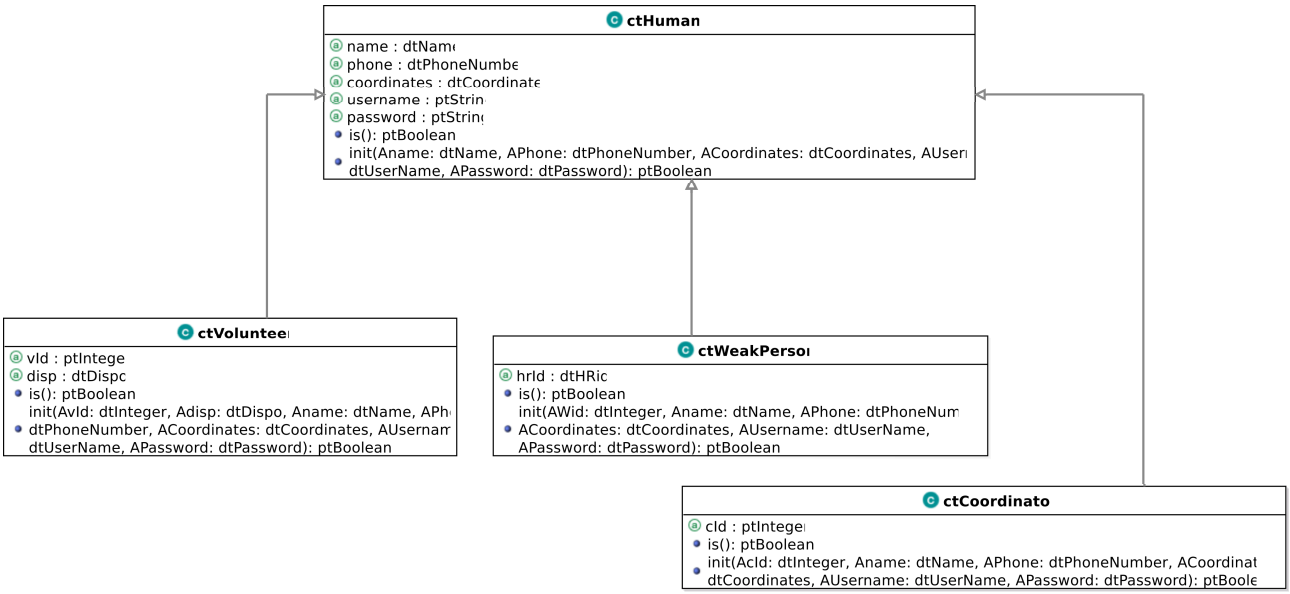


Figure 4.4: Concept Model - PrimaryTypes-Classes local view 05. Concept model local view.

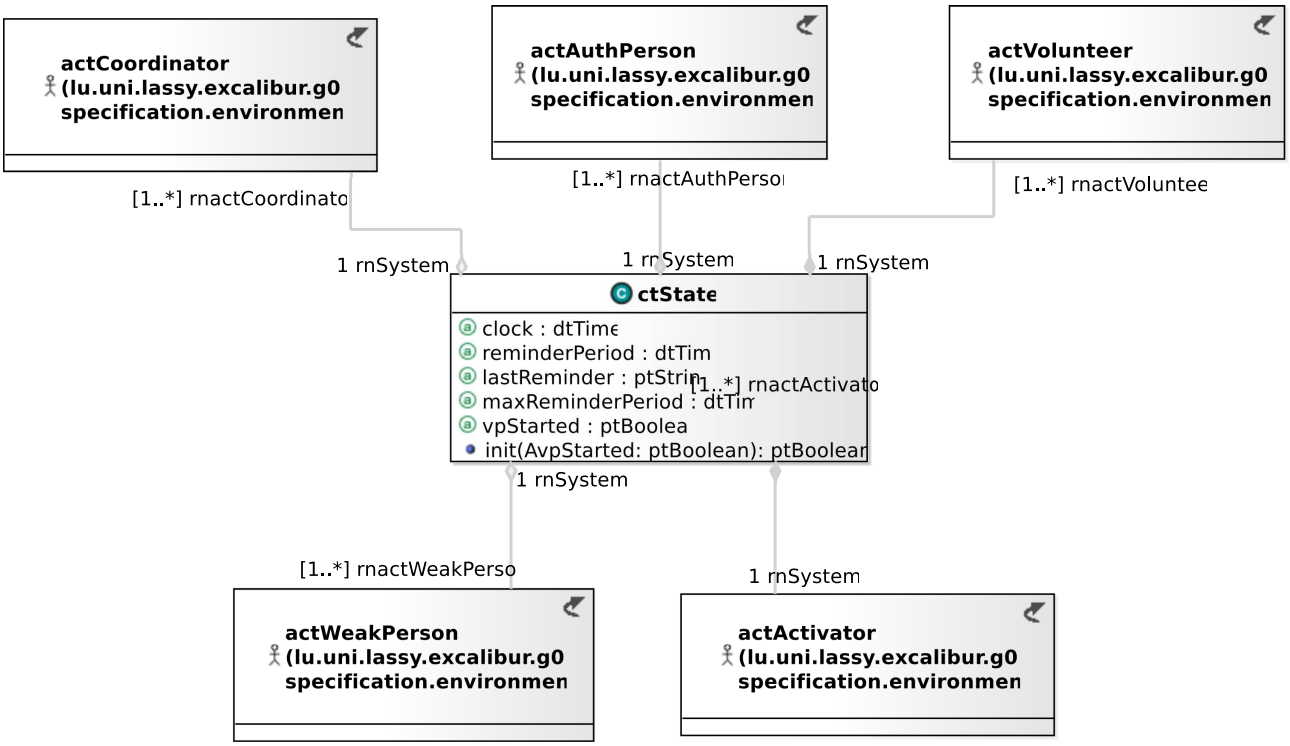


Figure 4.5: Concept Model - PrimaryTypes-Classes local view 07. Environment model global view.

Classes	
ctCheckBox	
used to model system's representation about checkbox answers from helprequest origin	
attribute	**q1: ptString** for ctCheckBox instance question 1
attribute	**q2: ptString** for ctCheckBox instance question 2
attribute	**q3: ptString** for ctCheckBox instance question 3
attribute	**q4: ptString** for ctCheckBox instance question 4
attribute	**q5: ptString** for ctCheckBox instance question 5
operation	**init():ptBoolean** Used to initialize the current object as a new instance of the ctCheckbox type
ctCoordinator	
Used to model the system representation about the actor that should handle help requests	
extends	lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.ctHuman
attribute	**cId: ptInteger** the coordinator instance unique ID
operation	**init(AcId:dtInteger, Aname:dtName, APhone:dtPhoneNumber, ACoordinates:dtCoordinates, AUsername:dtUserName, APassword:dtPassword):ptBoolean**

continues in next page . . .

. . . Classes table continuation

	Used to initialize the current object as a new instance of the ctCoordinator type
operation	**is():ptBoolean**
	used to determine if object is considered as a valid ctCoordinator
ctHelpRequest	
Used to model help requests sent any registered weak person having communication capability	
attribute	**handlingWaitingTime: dtTime**
	the time delta between current clock and time at which help request hqs been sent
attribute	**HrTime: dtTime**
	the time at which the help request has been sent
attribute	**priority: dtInteger**
	the priority set by coordinator after being called
operation	**init(AHrTime:dtTime):ptBoolean**
	Used to initialize the current object as a new instance of the ctHelpRequest type
operation	**is():ptBoolean**
	used to identify a system object as valid ctHelpRequest instance
ctHuman	
used to model system's representation about actors that shares the property of a human	
attribute	**coordinates: dtCoordinates**
	for ctHuman instance location
attribute	**name: dtName**
	for ctHuman instance name
attribute	**password: ptString**
	for ctHuman instance password
attribute	**phone: dtPhoneNumber**
	for ctHuman instance phone number
attribute	**username: ptString**
	for ctHuman instance username
operation	**init(Aname:dtName, APhone:dtPhoneNumber, ACoordinates:dtCoordinates, AUsername:dtUserName, APassword:dtPassword, AvpTokent:ptBoolean):ptBoolean**
	Used to initialize the current object as a new instance of the ctHuman type
operation	**is():ptBoolean**
	used to determine which object is considered as a valid ctHuman
ctState	
used to model the system. Each system specified using Messir must include a ctState class for which there is only one instance at any state of the abstract machine after creation.	
attribute	**clock: dtTime**
	used to represent system local time
attribute	**lastReminder: ptString**
attribute	**maxReminderPeriod: dtTime**
	used to set a maximum amount of time that should pass until notification is sent
attribute	**reminderPeriod: dtTime**
attribute	**vpStarted: ptBoolean**
	used to store the information of whether the system is started or not
operation	**init(AvpStarted:ptBoolean):ptBoolean**

continues in next page . . .

... Classes table continuation

	Used to initialize the current object as a new instance of the ctState type
ctVolunteer	
Used to model the system representation about the actor that should handle a low priority helprequest	
extends	lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.ctHuman
attribute	**disp: dtDispo**
	the volunteer disponibility
attribute	**handledHelpRequest: dtHRid**
	the volunteer handled help request ID
attribute	**vId: ptInteger**
	the volunteer unique ID
operation	**init(AvId:dtInteger, Adisp:dtDispo, Aname:dtName, APhone:dtPhoneNumber, ACoordinates:dtCoordinates, AUsername:dtUserName, APassword:dtPassword):ptBoolean**
	Used to initialize the current object as a new instance of the ctVolunteer type
operation	**is():ptBoolean**
	used to determine if object is considered as a valid ctVolunteer
ctWeakPerson	
Used to model the system representation about the actor that issue new help requests	
extends	lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.ctHuman
attribute	**hrId: dtHRid**
	unique weak person ID
operation	**init(AWid:dtInteger, Aname:dtName, APhone:dtPhoneNumber, ACoordinates:dtCoordinates, AUsername:dtUserName, APassword:dtPassword):ptBoolean**
	Used to initialize the current object as a new instance of the ctWeakPerson type
operation	**is():ptBoolean**
	used to determine if object is considered as a valid ctWeakPerson

4.2.2 Primary types - Datatypes types descriptions

The table below is providing comments on the graphical views given for the datatype types of the primary types.

DATATYPES	
dtAddress	
An integer used to store Address value	
attribute	**city: ptString**
	for the city name of the address
attribute	**num: ptInteger**
	for the location number of the address
attribute	**street: ptString**
	for the street name of the address
dtCoordinates	
used to identify position of geogrphical position on earth, defined as couple of longitude and latitude	
attribute	**lat: ptReal**
	for the latitude of the coordinates
attribute	**long: ptReal**
	for the longitude of the coordinates

continues in next page ...

. . . Datatypes table continuation

operation	**is():ptBoolean** used to determine which string is considered as a valid coordinate
dtDispo	
used to define dispos over a day	
attribute	**time: dtDateAndTime** for time representing dispos time
dtHRid	
Used to define a unique helprequest id	
attribute	**value: ptInteger** An integer used to identify a helpRequest
operation	**is():ptBoolean** used to determine if an integer is a correct helprequest
dtMessage	
A string with a message	
attribute	**MessageText: ptString** A message as a string
operation	**is():ptBoolean** used to determine which strings are considered as valid for a message
dtName	
A string used to identify names	
attribute	**value: ptString** for the string value of the name
operation	**is():ptBoolean** used to determine if the string is considered as validated
dtPassword	
A string used to identify password	
attribute	**value: ptString** for the value as a string
operation	**is():ptBoolean** used to determine which strings are considered as valid
dtPhoneNumber	
A string used to store phone number value	
attribute	**value: ptInteger** for the value of the phone number
operation	**is():ptBoolean** used to determine which integers are considered valid as phone number
dtRange	
An integer used to store range value	
attribute	**value: ptInteger** for the value of the range
operation	**is():ptBoolean** used to determine which integers are considered as a valid range
dtUserName	
A string to store the username	
attribute	**value: ptString** for storing the username
operation	**is():ptBoolean**

continues in next page . . .

. . . Datatypes table continuation

used to determine which strings are considered valid as usernames

Enumerations
etHelpRequestStatus An enumeration of the statuses of the help requests which can be: not processed, processed, not handled or handled
etVolunteerStatus An enumeration of the statuses of the volunteers which can be: offline, online or occupied

4.2.3 Primary types - Association types descriptions

There are no association types for the primary types.

4.2.4 Primary types - Aggregation types descriptions

There are no aggregation types for the primary types.

4.2.4.1 Primary types - Composition types descriptions

There are no composition types for the primary types.

4.2.5 Secondary types - Class types descriptions

There are no elements in this category in the system analysed.

4.2.6 Secondary types - Datatypes types descriptions

There are no elements in this category in the system analysed.

4.2.7 Secondary types - Association types descriptions

There are no association types for the secondary types.

4.2.8 Secondary types - Aggregation types descriptions

There are no aggregation types for the secondary types.

4.2.9 Secondary types - Composition types descriptions

There are no composition types for the secondary types.

Chapter 5

Operation Model

This section contains the operation schemes of each operation defined in either an actor, its output interface, in a primary or secondary type (class, datatype or enumeration types). The Messir OCL code listing is joined to the comment table.

5.1 Environment - Out Interface Operation Scheme for actActivator

5.1.1 Operation Model for oeSendNotificationToVolunteer

The `oeSendNotificationToVolunteer` operation has the following properties:

Operation	
oeSendNotificationToVolunteer[proactive]	
A proactive message used to avoid pending help request to stay too long in a not handle status	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	System is started
PreP 2	There are instance of type ctHelpRequest which have their attribute handlingWaitingTime greater or equal to 10 minutes, their attribute priority equal to 3 and instances of type ctVolunteer with attribute volunteerStatus equal to Online
Pre-Condition (functional)	
PreF 1	none
Post-Condition (functional)	
PostF 1	Operation sendNotification is triggered for all instance of ctVolunteer with status Online
PostF 2	
Post-Condition (protocol)	
PostP 1	

5.1.2 Operation Model for oeSetClock

The `oeSetClock` operation has the following properties:

Operation
oeSetClock[proactive]
An active message used to statically set the date and time information in the system's state.

continues in next page . . .

. . . Operation table continuation

Parameters	
1	**AcurrentClock: dtDateAndTime** The current date and time.
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	The system is created and started and the actual clock value is greater than @pre Clock value of ctState
Pre-Condition (functional)	
PreF 1	
Post-Condition (functional)	
PostF 1	The @Post instance of ctState has his clock variable correspond to the actual date and time
Post-Condition (protocol)	
PostP 1	

5.2 Environment - Out Interface Operation Scheme for actCoordinator

5.2.1 Operation Model for oeConfirmPriority

The `oeConfirmPriority` operation has the following properties:

Operation	
oeConfirmPriority Confirm the priority that the system has calculated	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the oeSendFilledCheckbox() operation has been called previously.
PreP 2	the corresponding ctHelpRequest still exists
Pre-Condition (functional)	
PreF 1	at least one item from ctCheckbox instance has been selected
Post-Condition (functional)	
PostF 1	the risk level of ctWeakPerson with corresponding dtHRid is set to the calculated level
Post-Condition (protocol)	
PostP 1	the ctHelpRequest instance of ctWeakPerson will appear to all near ctVolunteer by calling the operation oeGetMissionInRange()

5.2.2 Operation Model for oeGetHelpRequestDetail

The `oeGetHelpRequestDetail` operation has the following properties:

Operation
oeGetHelpRequestDetail Get the details of the help request by requiring the ID of the request
Parameters

continues in next page . . .

. . . Operation table continuation

1	**AdtHelpRequestId: dtHRid**
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the actor ctCoordinator is logged in
PreP 2	the selected ctWeakPerson instance must exist
Pre-Condition (functional)	
PreF 1	the dtHRid should exist and it is not null
Post-Condition (functional)	
PostF 1	a list with all the details of ctWeakPerson with the dtHRid is generated
Post-Condition (protocol)	
PostP 1	

5.2.3 Operation Model for oeGetPendingHelpRequests

The `oeGetPendingHelpRequests` operation has the following properties:

OPERATION	
oeGetPendingHelpRequests Generate a list with pending help requests with a certain status	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the actor ctCoordinator must be logged in
Pre-Condition (functional)	
PreF 1	there must exist ctWeakPerson instance with a certain etHelpRequestStatus attribute
Post-Condition (functional)	
PostF 1	generate a list with all ctWeakPerson instances that has the same etHelpRequestStatus attribute
Post-Condition (protocol)	
PostP 1	The operation oeReqCall() is available for all ctWeakPerson instances with a etHelpRequestStatus set to: not processed

5.2.4 Operation Model for oeProceedCall

The `oeProceedCall` operation has the following properties:

OPERATION	
oeProceedCall Proceed to call the weak person	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the ctWeakPerson instance with dtPhoneNumber exists
Pre-Condition (functional)	

continues in next page . . .

... Operation table continuation

PreF 1	the ctPhoneCompany instance has send the confirmation that the dtPhoneNumber of the corresponding ctWeakPerson instance exists
Post-Condition (functional)	
PostF 1	
Post-Condition (protocol)	
PostP 1	The operation oeReqCheckbox() is available

5.2.5 Operation Model for oeReqCheckbox

The `oeReqCheckbox` operation has the following properties:

OPERATION	
oeReqCheckbox	
Request a Checkbox to enter the illness information of the weak person	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the ctWeakPerson instance with dtHRid exists
PreP 2	the subfunction oeProceedCall() has been previously activated
Pre-Condition (functional)	
PreF 1	
Post-Condition (functional)	
PostF 1	
Post-Condition (protocol)	
PostP 1	the ctCoordinator receives ctCheckbox instance with ptString attributes

5.2.6 Operation Model for oeReqCall

The `oeReqCall` operation has the following properties:

OPERATION	
oeReqCall	
Request a call from phone company	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the ctWeakPerson instance with dtHRid exists
Pre-Condition (functional)	
PreF 1	The value of the attribute HelpRequestStatus of ctHelpRequest instance corresponding to AdtID : Not processed
PreF 2	The value of the attribute priority of ctHelpRequest instance is 0
Post-Condition (functional)	
PostF 1	the ctPhoneCompany instance received the request
Post-Condition (protocol)	
PostP 1	

5.3 Environment - Out Interface Operation Scheme for actVolunteer

5.3.1 Operation Model for oeSendNotification

The `oeSendNotification` operation has the following properties:

Operation	
oeSendNotification[proactive]	
This proactive operation is triggered by the activator is used to notificate the instance of volunteer that there pending mission is his range	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	Attribute vpStarted from instance of ctState is true
PreP 2	The attribute vpToken from instance of ctVolunteer is true
PreP 3	The attribute volunteerStatus of this instance of ctVolunteer is Online
PreP 4	There exists instance of ctHelpRequest with attribute Coordinates within the range of the instance of ctVolunteer and attributes priority > 2 and helpRequestStatus attribute = Not handled
Pre-Condition (functional)	
PreF 1	A notification is send to the volunteer
Post-Condition (functional)	
PostF 1	
Post-Condition (protocol)	
PostP 1	

5.3.2 Operation Model for oeAcceptMission

The `oeAcceptMission` operation has the following properties:

Operation	
oeAcceptMission	
Sent to system by the Volunteer to notify handling of the instance of ctHelpRequest corresponding to AdtID .	
Parameters	
1	**AdtId: dtHRid** First information that represent the Help request ID selected by the volunteer
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	The oeGetMissionInRAnge() has been executed and return at least 1 instance of ctHelpRequest.
PreP 2	The instance of ctVolunteer has its vpToken to true
Pre-Condition (functional)	
PreF 1	The value of the attribute HelpRequestStatus of the instance of ctHelpRequest corresponding to AdtID : Not Handled
PreF 2	The value of the attribute priority of the instance of ctHelpRequest is 3
Post-Condition (functional)	

continues in next page . . .

... Operation table continuation

PostF 1	The attribute HelpRequestStatus of the instance of ctHelpRequest is set to Handled
PostF 2	The attribute handledHelpRequest of the instance of ctVolunteer is set to AdtID
PostF 3	The attribute handlingWaitingTime of the instance of ctHelpRequest is set to null
Post-Condition (protocol)	
PostP 1	The operation oeEndMission() is available

5.3.3 Operation Model for oeGetPosition

The `oeGetPosition` operation has the following properties:

OPERATION	
oeGetPosition	
This allows the volunteer to request coordinates to the actSensor	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	
Pre-Condition (functional)	
PreF 1	
Post-Condition (functional)	
PostF 1	Returned coordinates form the sensor are coherent (are actual earth coordinates)
Post-Condition (protocol)	
PostP 1	

5.3.4 Operation Model for oeGetMissionInRagne

The `oeGetMissionInRagne` operation has the following properties:

OPERATION	
oeGetMissionInRagne	
Allows the volunteer to retrieve instances of ctHelpRequest matching his Range attributes	
Parameters	
1	**AdtRange: dtRange**
	First information used to specify the range at which the system should filter the help request
2	**AdtPosition: dtCoordinates**
	Second information that represent the center around which the range will be applied
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	Attribute vpStarted from instance of ctState is true
PreP 2	The attribute vpToken from instance of ctVolunteer is true
Pre-Condition (functional)	
PreF 1	AdtPosition is a within Luxembourg boundaries
PreF 2	AdtRange is less or egal to 50
PreF 3	the exists instance of ctHelpRequest with attribute HelpRequestStatus egal to not handled
PreF 4	the exists instance of ctHelpRequest with attribute priority egal to 3

continues in next page ...

. . . Operation table continuation

Post-Condition (functional)	
PostF 1	
Post-Condition (protocol)	
PostP 1	The operation oeGetMissionDetails() is available

5.3.5 Operation Model for oeRegisterCoordinates

The `oeRegisterCoordinates` operation has the following properties:

OPERATION	
oeRegisterCoordinates	
This allow a specific volunteer to register his position in the system, this position will be the reference until he calls again or trigger oeGetMissionInrange()	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	Attribute vpStarted from instance of ctState is true
PreP 2	The attribute vpToken from instance of ctVolunteer is true
Pre-Condition (functional)	
PreF 1	AdtPosition are within Luxembourg Boundaries
Post-Condition (functional)	
PostF 1	Attribute dtCoordinates from the instance of ctVolunteer is set to dtCoordinates
Post-Condition (protocol)	
PostP 1	

5.4 Environment - Out Interface Operation Scheme for actWeakPerson

5.4.1 Operation Model for oeAlertFamily

The `oeAlertFamily` operation has the following properties:

OPERATION	
oeAlertFamily	
Alert the family members/relatives of the actor.	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	The coordinates attributes of the corresponding ctWeakPerson has been received.
PreP 2	The ctHelpRequest instance of the corresponding ctWeakPerson has been accepted by the system.
Pre-Condition (functional)	
PreF 1	The ctHelpRequest instance has name and phone number attributes of the corresponding ctWeakPersonFamily instances.
Post-Condition (functional)	

continues in next page . . .

. . . Operation table continuation

PostF 1	The ctPhoneCompany instance will receive an SMS for the corresponding ctWeakPersonFamily instances
Post-Condition (protocol)	
PostP 1	

5.5 Environment - Actor Operation Schemes

There are no elements in this category in the system analysed.

5.6 Primary Types - Operation Schemes for Classes

There are no elements in this category in the system analysed.

5.7 Primary Types - Operation Schemes for Datatypes

There are no elements in this category in the system analysed.

5.8 Primary Types - Operation Schemes for Enumerations

There are no elements in this category in the system analysed.

5.9 Secondary Types - Operation Schemes for Classes

There are no elements in this category in the system analysed.

5.10 Secondary Types - Operation Schemes for Datatypes

There are no elements in this category in the system analysed.

5.11 Secondary Types - Operation Schemes for Enumerations

There are no elements in this category in the system analysed.

Chapter 6

Test Model(s)

There are no elements in this category in the system analysed.

Chapter 7

Additional Constraints

Appendix A

Undocumented Messir Specification Elements

A.1 Undocumented Use Cases

A.1.1 Undocumented Summary Level Use Cases

- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.suAltertAFamilyMember
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.ugRequestHelp

A.1.2 Undocumented Subfunction Level Use Cases

- lu.uni.lassy.excalibur.g01.specification.usecases.volunteer.oeGetCurrentPosition
- lu.uni.lassy.excalibur.g01.specification.usecases.coordinator.oeGetHelpRequestDetail
- lu.uni.lassy.excalibur.g01.specification.usecases.volunteer.oeGetInRangeMission
- lu.uni.lassy.excalibur.g01.specification.usecases.volunteer.oeGetMissionDetails
- lu.uni.lassy.excalibur.g01.specification.usecases.volunteer.oeRegisterPosition
- lu.uni.lassy.excalibur.g01.specification.usecases.volunteer.oeSendCurrentPosition
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.oeAlertFamily
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.oeGetPositionFromSensor
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.oeInputManualPos
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.oeSendDeliveryReport
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.oeSendHelpRequest
- lu.uni.lassy.excalibur.g01.specification.usecases.activator.oeSendNotificationToVolunteer
- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.oeSendPos
- lu.uni.lassy.excalibur.g01.specification.usecases.activator.oeSetClock

A.2 Undocumented Use Case Instances

A.2.1 Undocumented Summary Level Use Case Instances

- lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.uciAlertTheFamily

A.3 Undocumented Environment Model Views

- em-view8

A.4 Undocumented Primary Relationships

A.4.1 Undocumented Primary Type Associations

- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.assActCoordinatorVolunteer
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.assCheckBWeakPerson
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.assCoordinatorHelpRequest
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.assHelpRequest
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.assVolunteerHR

A.5 Undocumented Secondary Types

A.5.1 Undocumented Secondary Datatype Types

- lu.uni.lassy.excalibur.g01.specification.concepts.secondarytypes.datatypes.dtTime

A.6 Undocumented Operation Specifications

- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.ctCoordinator.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.ctHelpRequest.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes.ctVolunteer.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtCoordinates.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtHRid.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtMessage.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtName.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtPassword.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtPhoneNumber.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtRange.is
- lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes.dtUserName.is
- lu.uni.lassy.excalibur.g01.specification.environment.actPhoneCompany.outactPhoneCompany.oeGetConfirm

- lu.uni.lassy.excalibur.g01.specification.environment.actPhoneCompany.outactPhoneCompany.oeSendDeliveryRep
- lu.uni.lassy.excalibur.g01.specification.environment.actPositionInputActor.outactPositionInputActor.oeInputPost
- lu.uni.lassy.excalibur.g01.specification.environment.actPositionRequester.outactPositionRequester.oeGetPositionI
- lu.uni.lassy.excalibur.g01.specification.environment.actWeakPerson.outactWeakPerson.oeGetpostition
- lu.uni.lassy.excalibur.g01.specification.environment.actWeakPersonFamily.outactWeakPersonFamily.oeConfirmCa
- lu.uni.lassy.excalibur.g01.specification.environment.actWeakPersonFamily.outactWeakPersonFamily.oeConfirmM
- lu.uni.lassy.excalibur.g01.specification.environment.actWeakPersonFamily.outactWeakPersonFamily.oeSubscribe

Appendix B

Messir Specification Files Listing

B.1 File ./src-gen/messir-spec/.views.msr

```
//
//DON'T TOUCH THIS FILE !!!
//
package uuid5a859098d2dc4161afdb7ab26ea5a813 {
 Concept Model {}
}
```

Listing B.1: Messir Spec. file .views.msr.

B.2 File ./src-gen/messir-spec/usecases/activatorUC.msr

```
/*
* @author Adriano
* @date Sun Dec 04 14:54:42 CET 2016
*/

package lu.uni.lassy.excalibur.g01.specification.usecases.activator {

import lu.uni.lassy.messir.libraries.math
import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.string
import lu.uni.lassy.excalibur.g01.specification.environment
import lu.uni.lassy.excalibur.g01.specification.usescases.weakperson
import lu.uni.lassy.excalibur.g01.specification.usecases.coordinator
import lu.uni.lassy.excalibur.g01.specification.usecases.volunteer

 Use Case Model {

  use case system subfunction oeSetClock() {
   actor actActivator[primary,proactive]
  }
  use case system subfunction oeSendNotificationToVolunteer () {
   actor actActivator[primary,proactive]
   actor actVolunteer[secondary,passive]

   returned messages{
    ieMessage(adtMessage) returned to actVolunteer
   }
  }

 }

}
```

Listing B.2: Messir Spec. file activatorUC.msr.

B.3 File ./src-gen/messir-spec/usecases/DeployAndRun.msr

```
/*
* @author Adriano
* @date Sun Dec 04 15:09:10 CET 2016
*/

package lu.uni.lassy.excalibur.g01.specification.usecases.main {
import lu.uni.lassy.messir.libraries.math
import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.string
import lu.uni.lassy.excalibur.g01.specification.environment
import lu.uni.lassy.excalibur.g01.specification.usescases.weakperson
import lu.uni.lassy.excalibur.g01.specification.usecases.coordinator
import lu.uni.lassy.excalibur.g01.specification.usecases.volunteer
import lu.uni.lassy.excalibur.g01.specification.usecases.activator

 Use Case Model {

  use case instance uciNormalRun : suDeployRun {
   actors {
    activator : actActivator
    Bob : actCoordinator
    Benny : actWeakPerson
    Mike : actVolunteer
    GPS : actPositionRequester
    Garmin : actSensor
    Tango : actPhoneCompany

   }
   use case steps {
    activator executed instanceof subfunction oeSetClock("12:00:00 13.09.16") {

     }
    Benny executed instanceof ugRequestHelp() {
      use case steps {
      GPS executed instanceof subfunction oeGetPositionFromSensor() {
      iereqCurrentPos() returned to Garmin
     }
    Garmin executed instanceof subfunction oeSendPos("49.626506, 6.163065") {
      ieSentdPosition("49.626506, 6.163065") returned to Benny
     }
    Benny executed instanceof subfunction oeInputManualPos("1 rue des maraîchers-Luxembourg", "
        49.626506, 6.163065") {
      ieConfirmPos("1 rue des maraîchers-Luxembourg", "49.626506, 6.163065") returned to Benny
     }
    Benny executed instanceof subfunction oeSendHelpRequest("Benny Hill" , "6611726354", "49.618416,
         6.172678" , "1 rue de Cents - Luxembourg") {
      ieSentConfirmationMessage("Your help request has been sent") returned to Benny
     }

    Benny executed instanceof subfunction oeAlertFamily("Hid : 12") {
       ieSmsForFamily("Benny has sent an help request","6611726354") returned to Tango
     }
      }
     }

    Bob executed instanceof ugRetrievePendingHelpRequestDetails() {
      use case steps {
      Bob executed instanceof subfunction oeGetPendingHelpRequests() {
      ieSendPendingHR("12h05","661234567") returned to Bob
      ieSendPendingHR("12h06","661334569") returned to Bob
      ieSendPendingHR("12h09","691007221") returned to Bob

     }

    Bob executed instanceof subfunction oeGetHelpRequestDetail("id : 32") {
     ieSendHelpRequestDetail("Paul Kremmer" , "6611726354", "49.618416, 6.172678" , "1 rue de Cents
         - Luxembourg") returned to Bob

```

```
66      }
67
68       }
69      }
70     Bob executed instanceof suCallSelectedHelpRequest() {
71       use case steps {
72        Bob executed instanceof subfunction oeReqCall("661234567"){
73       ieConfirm("661234567") returned to Tango
74      }
75
76       }
77      }
78     Bob executed instanceof ugAssignPriorityToHelpRequest() {
79       use case steps {
80        Bob executed instanceof subfunction oeReqCheckbox(){
81       ieSendCheckbox("Speaks slowly","Needs Water","Conscious") returned to Bob
82      }
83        Bob executed instanceof subfunction oeSendFilledCheckbox("No","Yes","Yes"){
84      ieSendCalculatedPriority("The calculated priority is: Low") returned to Bob
85     }
86
87        Bob executed instanceof subfunction oeConfirmPriority("OK"){
88      ieSendResult("Urgent") returned to Bob
89     }
90
91       }
92      }
93
94     activator executed instanceof subfunction oeSetClock("12:10:00 13.09.16") {
95
96      }
97     activator executed instanceof subfunction oeSendNotificationToVolunteer() {
98      ieMessage("There are Mission in your specified range that must be handled") returned to Mike
99      }
100    Mike executed instanceof ugGetMissionInRange() {
101      use case steps {
102
103    GPS executed instanceof subfunction oeGetPositionFromSensor() {
104      iereqCurrentPos() returned to Garmin
105     }
106    Garmin executed instanceof subfunction oeSendPos("49.626506, 6.163065") {
107      ieSentCurrentPosition("49.626506, 6.163065") returned to Mike
108     }
109    Mike executed instanceof subfunction oeGetInRangeMission("25","49.63999, 6.16399") {
110     ieSendInRangeMission("id=29","49.626506, 6.163065") returned to Mike
111     ieSendInRangeMission("id=34","49.627499, 6.274156") returned to Mike
112     ieSendInRangeMission("id=18","49.639006, 6.263164") returned to Mike
113
114     }
115
116      }
117     }
118    Mike executed instanceof suRetrieveMissionDetails() {
119      use case steps {
120
121      }
122     }
123    Mike executed instanceof suAcceptMission() {
124      use case steps {
125
126      }
127     }
128
129   }
130  }
131
132 use case system summary suDeployRun() {
133  actor actCoordinator[primary,active]
134  actor actVolunteer[primary,active]
135  actor actWeakPerson[primary,active]
```

```
136   actor actActivator[primary,proactive]
137
138   step a: actWeakPerson executes ugRequestHelp()
139   step b :actCoordinator executes ugRetrievePendingHelpRequestDetails()
140   step c :actCoordinator executes suCallSelectedHelpRequest()
141   step d :actCoordinator executes ugAssignPriorityToHelpRequest()
142   step e :actActivator executes oeSetClock()
143   step f :actActivator executes oeSendNotificationToVolunteer()
144   step g :actVolunteer executes ugGetMissionInRange()
145   step h :actVolunteer executes suRetrieveMissionDetails()
146   step i :actVolunteer executes suAcceptMission()
147
148  }
149
150  }
151
152 }
```

Listing B.3: Messir Spec. file DeployAndRun.msr.

B.4 File ./src-gen/messir-spec/operations/environment/environment-actActivator-oeSendNotificationToVolunteer.msr

```
1 package lu.uni.lassy.excalibur.g01.specification.environment.operations.actActivator.outactActivator
      .oeSendNotificationToVolunteer {
2
3 import lu.uni.lassy.messir.libraries.primitives
4 import lu.uni.lassy.messir.libraries.math
5 import lu.uni.lassy.messir.libraries.string
6 import lu.uni.lassy.messir.libraries.calendar
7 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes
8 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes
9
10  Operation Model {
11    operation: lu.uni.lassy.excalibur.g01.specification.environment.actActivator.outactActivator.
          oeSendNotificationToVolunteer():ptBoolean{
12    preP{
13     let System: ctState in
14     let AvpStarted: ptBoolean in
15     let NotHandledHelpRequest : Bag(ctHelpRequest) in
16     let FreeVolunteer : Bag(ctVolunteer) in
17
18     //PreP01
19     self.rnActor.rnSystem = System
20     and System.vpStarted = AvpStarted
21     and AvpStarted = true
22
23     //Prep02
24     and System.rnctHelpRequest -> select(
25      System.rnctHelpRequest.HelpRequestStatus = "Not handled"
26      and System.rnctHelpRequest.priority > 2
27     )
28     = NotHandledHelpRequest
29
30     and System.rnctHuman.rnctVolunteer -> select(
31      System.rnctHuman.rnctVolunteer.VolunteerStatus = "Online"
32     )
33     = FreeVolunteer
34
35     and NotHandledHelpRequest -> Size().geq(1)
36     and FreeVolunteer -> Size().geq(1)
37
38    }preF{
39     true
40    }postF{
41       FreeVolunteer->forall(rnInterfaceOUT^oeSendNotification())
42
43    }postP{
```

```
44
45    }
46
47   }
48
49   }
50  }
```

Listing B.4: Messir Spec. file environment-actActivator-oeSendNotificationToVolunteer.msr.

B.5 File ./src-gen/messir-spec/operations/environment/environment-actVolunteer-oeGetPosition.msr

```
1 package lu.uni.lassy.excalibur.g01.specification.environment.operations.actVolunteer.outactVolunteer
      .oeGetPosition {
2
3 import lu.uni.lassy.messir.libraries.primitives
4 import lu.uni.lassy.messir.libraries.math
5 import lu.uni.lassy.messir.libraries.string
6 import lu.uni.lassy.messir.libraries.calendar
7
8  Operation Model {
9
10  }
11 }
```

Listing B.5: Messir Spec. file environment-actVolunteer-oeGetPosition.msr.

B.6 File ./src-gen/messir-spec/operations/environment/environment-actVolunteer-oeLogin.msr

```
1 package lu.uni.lassy.excalibur.g01.specification.environment.operations.actVolunteer.outactVolunteer
      .oeLogin {
2
3 import lu.uni.lassy.messir.libraries.primitives
4 import lu.uni.lassy.messir.libraries.math
5 import lu.uni.lassy.messir.libraries.string
6 import lu.uni.lassy.messir.libraries.calendar
7
8  Operation Model {
9
10  }
11 }
```

Listing B.6: Messir Spec. file environment-actVolunteer-oeLogin.msr.

B.7 File ./src-gen/messir-spec/operations/environment/environment-actVolunteer-oeSendNotification.msr

```
1 package lu.uni.lassy.excalibur.g01.specification.environment.operations.actVolunteer.outactVolunteer
      .oeSendNotification {
2
3 import lu.uni.lassy.messir.libraries.primitives
4 import lu.uni.lassy.messir.libraries.math
5 import lu.uni.lassy.messir.libraries.string
6 import lu.uni.lassy.messir.libraries.calendar
7 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes
8 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes
9
10  Operation Model {
11   operation: lu.uni.lassy.excalibur.g01.specification.environment.actVolunteer.outactVolunteer.
        oeSendNotification():ptBoolean{
12    preP{
13     let System: ctState in
```

```
14     let AvpStarted: ptBoolean in
15     let NotHandledHelpRequest : Bag(ctHelpRequest) in
16     let volunteer : ctVolunteer in
17     let vCoordinates : dtCoordinates in
18     let vRange : dtRange in
19     let inRangeHelpRequest : Bag(ctHelpRequest) in
20
21     //PreP01
22     self.rnActor.rnSystem = System
23     and System.vpStarted = AvpStarted
24     and AvpStarted = true
25
26     //Prep02
27     and volunteer =System.rnActor
28     and volunteer.rnctHuman.vpToken = true
29
30     //Prep03
31     and volunteer.volunteerStatus = 'Online'
32
33     //Prep04
34     and System.rnctHelpRequest -> select(
35      System.rnctHelpRequest.HelpRequestStatus = "Not handled"
36      and System.rnctHelpRequest.priority > 2
37     )
38     = NotHandledHelpRequest
39     and NotHandledHelpRequest -> Size().geq(1)
40
41     and vRange = volunteer.range
42     and vCoordinates = vlounteer.rnctHuman.coordinates
43     and NotHandledHelpRequest -> iterate(hr : ctHelpRequest ; acc : dtHRid = 0 |
           NotHandledHelpRequest -> select(acc=hrid) = thisHelpRequest
44                         let hrCoordinates : dtCoordinates = thisHelpRequest.rnctHelpRequest.rnctHuman
                               .coordinates in
45                         and volunteer.rnctHuman.isInRange (vCoordinates,vRange,hrCoordinates)
46
47     ) = inRangeHelpRequest
48
49     and inRangeHelpRequest -> size().geq(1)
50
51    }
52    preF{
53     true
54    }
55    postF{
56     rnInterfaceIN^ieMessage("There exists pending help request in near "+vRange+" km")
57    }
58
59   }
60
61  }
62 }
```

Listing B.7: Messir Spec. file environment-actVolunteer-oeSendNotification.msr.

B.8 File ./src-gen/messir-spec/environment/environment.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.environment {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes
13 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes
```

```
14
15 Environment Model {
16
17  actor actAuthPerson role rnactAuthPerson cardinality [1..*] {
18
19   input interface inactAuthPerson {
20   operation ieMessage(AdtMessage : dtString) : ptBoolean
21   }
22   output interface outactAuthPerson {
23   operation oeLogin(AdtUsr : dtUserName, AdtPsw : dtPassword) : ptBoolean
24   operation oeLogout() : ptBoolean
25   }
26  }
27
28  actor actActivator role rnactActivator cardinality [1..*] {
29
30   input interface inactActivator {
31
32   }
33
34   output interface outactActivator {
35    proactive operation oeSetClock(AcurrentClock : dtDateAndTime):ptBoolean
36    proactive operation oeSendNotificationToVolunteer():ptBoolean
37
38   }
39  }
40
41  actor actMsrCreator role rnactMsrCreator cardinality [1..*] {
42
43   input interface inactMsrCreator {
44
45   }
46   output interface outactMsrCreator {
47   operation oeCreateSystemAndEnvironment() : ptBoolean
48   }
49  }
50
51  actor actSensor role rnactSensor cardinality [1..*] {
52
53   input interface inactSensor {
54    operation iereqCurrentPos() : ptBoolean
55    operation ieReturnCurentPosition(adtCoordinates : dtCoordinates) : ptBoolean
56   }
57   output interface outactSensor {
58   operation oeSendPos():ptBoolean
59   }
60  }
61
62  actor actPositionInputActor role rnactPositionInputActor cardinality [1..*] {
63
64   input interface inactPositionInputActor {
65    operation ieSentPosition() : ptBoolean
66   }
67   output interface outactPositionInputActor {
68    operation oeInputPost() : ptBoolean
69   }
70  }
71
72  actor actCoordinator role rnactCoordinator cardinality [1..*] {
73
74   input interface inactCoordinator {
75    operation ieSendPendingHelpRequestList(AdtTime:dtTime,AdtPhoneNumber :dtPhoneNumber) : ptBoolean
76    operation ieSendHelpRequestDetail(AdtName : dtName, AdtPhoneNumber : dtPhoneNumber ,
          AdtCoordinates : dtCoordinates , AdtAddress : dtAddress) : ptBoolean
77    operation ieSendVolunteerList(AetVolunteerStatus : etVolunteerStatus) : ptBoolean
78    operation ieConfirm(AdtPhoneNumber : dtPhoneNumber) : ptBoolean
79    operation ieSendCheckbox(ActCheckBox :ctCheckBox) : ptBoolean
80    operation ieSendCalculatedPriority(AdtRiskLevel : dtInteger, AdtHelpRequestID : dtHRid) :
          ptBoolean
81    operation ieSendResult(AdtRiskLevel : dtInteger, AdtHelpRequestID : dtHRid) : ptBoolean
```

```
82    operation ieReqPendingHelpRequests() : ptBoolean
83    operation ieRequHRDetails(AdtId :dtHRid) : ptBoolean
84    operation ieSendPendingHR(AdtTime : dtTime) : ptBoolean
85    operation ieFamilyDetailsRequest() : ptBoolean
86
87   }
88   output interface outactCoordinator {
89    operation oeGetPendingHelpRequests(AetHelpRequestStatus : etHelpRequestStatus) : ptBoolean
90    operation oeGetHelpRequestDetail(AdtHelpRequestId : dtHRid) : ptBoolean
91    operation oeProceedCall(AdtPhoneNumber : dtPhoneNumber) : ptBoolean
92
93    operation oeGetVolunteersList(AetVolunteerStatus : etVolunteerStatus) : ptBoolean
94    operation oeReqCall(AdtPhoneNumber : dtPhoneNumber) : ptBoolean
95    operation oeReqCheckbox() : ptBoolean
96    operation oeSendFilledCheckbox(ActCheckBox : ctCheckBox) : ptBoolean
97    operation oeConfirmPriority(AdtRiskLevel : dtInteger, AdtHelpRequestID : dtHRid) : ptBoolean
98   }
99  }
100
101 actor actWeakPerson role rnactWeakPerson cardinality [1..*] {
102
103  input interface inactWeakPerson {
104   operation ieSentdPosition() : ptBoolean
105   operation ieSentConfirmationMessage() : ptBoolean
106   operation ieSendInfo() : ptBoolean
107   operation ieConfirmPos() : ptBoolean
108   operation ieSmsForFamily(AdtMessage : dtMessage, AdtPhoneNumber : dtPhoneNumber) : ptBoolean
109
110  }
111  output interface outactWeakPerson {
112   operation oeSendHelpRequest() : ptBoolean
113   operation oeGetpostition() : ptBoolean
114   operation oeInputManualPos() : ptBoolean
115   operation oeGetInfo() : ptBoolean
116   operation oeGetPositionFromSensor() : ptBoolean
117   operation oeAlertFamily() : ptBoolean
118  }
119 }
120
121 actor actVolunteer role rnactVolunteer cardinality [1..*] {
122
123  input interface inactVolunteer {
124   operation ieSentCurrentPosition(AdtCoordinates : dtCoordinates) : ptBoolean
125   operation ieSentMissionConfirmation() : ptBoolean
126   operation ieSendRange() : ptBoolean
127   operation ieSendInRangeMission(AdtId : dtHRid, AdtCoordinates : dtCoordinates) : ptBoolean
128   operation ieRequestPosition() : ptBoolean
129   operation ieMessage(adtMessage : dtString) : ptBoolean
130  }
131  output interface outactVolunteer {
132   operation oeGetPosition() : ptBoolean
133   operation oeRegisterCoordinates(AdtPosition : dtCoordinates) : ptBoolean
134   proactive operation oeSendNotification() : ptBoolean
135   operation oeGetMissionInRagne(AdtRange: dtRange,AdtPosition : dtCoordinates) : ptBoolean
136   operation oeAcceptMission(AdtId : dtHRid) : ptBoolean
137
138  }
139 }
140
141 actor actWeakPersonFamily role rnactWeakPersonFamily cardinality [1..*] {
142
143  input interface inactWeakPersonFamily {
144   operation ieSentPosition() : ptBoolean
145   operation ieGetMessage() : ptBoolean
146   operation ieGetCall(): ptBoolean
147
148  }
149  output interface outactWeakPersonFamily {
150
151   operation oeSubscribe() : ptBoolean
```

```
152    operation oeConfirmMessage() : ptBoolean
153    operation oeConfirmCall() : ptBoolean
154
155   }
156  }
157
158  actor actPhoneCompany role rnactPhoneCompany cardinality [1..*]{
159   input interface inactPhoneCompany{
160    operation ieRequestConfirm(AdtPhoneNumber : dtPhoneNumber) : ptBoolean
161    operation ieSmsForFamily(AdtMessage : dtMessage, AdtPhoneNumber : dtPhoneNumber) : ptBoolean
162    operation ieMessageReception(): ptBoolean
163   }
164   output interface outactPhoneCompany{
165    operation oeGetConfirm() : ptBoolean
166    operation oeSendDeliveryReport(AdtMessage : dtMessage, AdtPhoneNumber : dtPhoneNumber) :
           ptBoolean
167   }
168  }
169
170  actor actPositionRequester role rnactPositionRequester cardinality [1..*] {
171
172   input interface inactPositionRequester {
173    operation ieSendSensorPosition() : ptBoolean
174   }
175   output interface outactPositionRequester {
176    operation oeGetPositionFromSensor() : ptBoolean
177   }
178  }
179
180 }
181 }
```

Listing B.8: Messir Spec. file environment.msr.

B.9 File ./src-gen/messir-spec/operations/environment/operation$_o$eAcceptMissio

```
1 /*
2 * @author Adriano
3 * @date Fri Dec 16 12:22:20 CET 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.environement.^operation.actVolunteer.
      outactVolunteer.oeAcceptMission {
7
8 import lu.uni.lassy.messir.libraries.primitives
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.string
11 import lu.uni.lassy.messir.libraries.calendar
12 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes
13 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes
14
15  Operation Model {
16    operation: lu.uni.lassy.excalibur.g01.specification.environment.actVolunteer.outactVolunteer.
          oeAcceptMission(AdtId: dtHRid):ptBoolean{
17
18    preP
19    {
20    let System: ctState in
21    let volunteer : ctVolunteer in
22    let inRangeHelpRequest : Bag(ctHelpRequest) in
23    let selectedHelpRequest : ctHelpRequest in
24     //PreP01
25     and self.rnActor.rnSystem = System
26     and volunteer = System.rnActor
27     and volunteer.rnInterface^oeGetMissionInRange() = true
28     and inRangeHelpRequest -> size().geq(1)
29     //Prep02
30     and volunteer.rnctHuman.cpToken = true
31
```

```
32    }
33
34    preF
35    {
36     and System.HelpRequest()->forall(hrid = AdtId) = selectedHelpRequest
37     and "not handled" =selectedHelpRequest.helpRequestSatus
38     and 3 = selectedHelpRequest.priority
39
40    }
41
42    postF{
43
44     true
45
46    }
47
48    postP
49    {
50     //postP01
51     selectedHelpRequest.HelpRequestStatus@post = 'handled'
52     //postp02
53     and volunteer.handledHelpRequest = AdtId
54     //postp03
55     and selectedHelpRequest.handlingWaitingTime = 0
56
57    }
58
59   }
60  }
61
62 }
```

Listing B.9: Messir Spec. file operation$_o eAcceptMission.msr$.

B.10 File ./src-gen/messir-spec/operations/operational.msr

```
1 /*
2 * @author Adriano
3 * @date Mon Dec 05 16:32:52 CET 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.^operation {
7 import lu.uni.lassy.messir.libraries.primitives
8 import lu.uni.lassy.messir.libraries.math
9 import lu.uni.lassy.messir.libraries.string
10 import lu.uni.lassy.messir.libraries.calendar
11 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes
12 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes
13
14  Operation Model {
15   operation: lu.uni.lassy.excalibur.g01.specification.environment.actVolunteer.outactVolunteer.
         oeGetPosition():ptBoolean{
16
17   }
18
19   operation: lu.uni.lassy.excalibur.g01.specification.environment.actVolunteer.outactVolunteer.
         oeGetMissionInRagne(AdtRange: dtRange, AdtPosition: dtCoordinates):ptBoolean{
20
21   }
22
23   operation: lu.uni.lassy.excalibur.g01.specification.environment.actVolunteer.outactVolunteer.
         oeRegisterCoordinates(AdtPosition : dtCoordinates):ptBoolean{
24
25   }
26
27   operation: lu.uni.lassy.excalibur.g01.specification.environment.actActivator.outactActivator.
         oeSetClock(AcurrentClock: dtDateAndTime):ptBoolean{
28
29   }
```

```
30
31   operation: lu.uni.lassy.excalibur.g01.specification.environment.actWeakPerson.outactWeakPerson.
         oeAlertFamily():ptBoolean{
32
33   }
34
35   operation: lu.uni.lassy.excalibur.g01.specification.environment.actCoordinator.outactCoordinator.
         oeConfirmPriority(AdtRiskLevel : dtInteger, AdtHelpRequestID : dtHRid):ptBoolean{
36
37   }
38
39   operation: lu.uni.lassy.excalibur.g01.specification.environment.actCoordinator.outactCoordinator.
         oeGetHelpRequestDetail(AdtHelpRequestId: dtHRid):ptBoolean{
40
41   }
42
43   operation: lu.uni.lassy.excalibur.g01.specification.environment.actCoordinator.outactCoordinator.
         oeGetPendingHelpRequests(AetHelpRequestStatus : etHelpRequestStatus):ptBoolean{
44
45   }
46   operation: lu.uni.lassy.excalibur.g01.specification.environment.actCoordinator.outactCoordinator.
         oeProceedCall(AdtPhoneNumber : dtPhoneNumber):ptBoolean{
47
48   }
49   operation: lu.uni.lassy.excalibur.g01.specification.environment.actCoordinator.outactCoordinator.
         oeReqCheckbox():ptBoolean{
50
51   }
52
53   operation: lu.uni.lassy.excalibur.g01.specification.environment.actCoordinator.outactCoordinator.
         oeReqCall(AdtPhoneNumber : dtPhoneNumber):ptBoolean{
54
55   }
56
57  }
58
59 }
```

Listing B.10: Messir Spec. file operational.msr.

B.11 File ./src-gen/messir-spec/concepts/primarytypes-associations/primarytypes-associations.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.associations {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12
13 Concept Model {
14
15  Primary Types {
16
17  }
18 }
19 }
```

Listing B.11: Messir Spec. file primarytypes-associations.msr.

B.12 File ./src-gen/messir-spec/concepts/primarytypes-classes/primarytypes-classes.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.classes {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12 import lu.uni.lassy.excalibur.g01.specification.environment
13
14 import lu.uni.lassy.messir.libraries.primitives
15 import lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes
16 Concept Model {
17
18  Primary Types {
19
20   class ctCheckBox role rnctCheckBox cardinality [0..*] {
21    attribute q1: ptString
22    attribute q2: ptString
23    attribute q3: ptString
24    attribute q4: ptString
25    attribute q5: ptString
26
27    operation init() : ptBoolean
28   }
29
30   state class ctState {
31    attribute clock: dtTime
32    attribute reminderPeriod: dtTime
33    attribute lastReminder: ptString
34    attribute maxReminderPeriod: dtTime
35    attribute vpStarted: ptBoolean
36
37    operation init(AvpStarted:ptBoolean): ptBoolean
38   }
39
40   class ctHuman role rnctHuman cardinality [1..*]{
41
42    attribute name: dtName
43    attribute phone: dtPhoneNumber
44    attribute coordinates: dtCoordinates
45    attribute username: ptString
46    attribute password: ptString
47    attribute vptoken: ptBoolean
48    operation is() : ptBoolean
49    operation init(Aname : dtName, APhone : dtPhoneNumber, ACoordinates : dtCoordinates, AUsername :
          dtUserName,APassword :dtPassword ,AvpTokent :ptBoolean ) : ptBoolean
50    operation isInRange(AdtCenterCoordinates : dtCoordinates , Arange : ptInteger ,
          AdtDestinationCoordinates : dtCoordinates) : ptBoolean
51   }
52
53   class ctWeakPerson role rnctWeakPerson cardinality [1..*] extends ctHuman {
54    attribute hrId: dtHRid
55
56    operation is() : ptBoolean
57    operation init(AWid : dtInteger,Aname : dtName, APhone : dtPhoneNumber, ACoordinates :
          dtCoordinates, AUsername : dtUserName,APassword :dtPassword) : ptBoolean
58   }
59
60   class ctCoordinator role rnctCoordinator cardinality [1..*] extends ctHuman {
61    attribute cId: ptInteger
62    operation is() : ptBoolean
63    operation init(AcId : dtInteger,Aname : dtName, APhone : dtPhoneNumber, ACoordinates :
```

```
        dtCoordinates, AUsername : dtUserName,APassword :dtPassword) : ptBoolean
64  }
65
66  class ctVolunteer role rnctVolunteer cardinality [1..*] extends ctHuman {
67   attribute vId: ptInteger
68   attribute disp: dtDispo
69   attribute handledHelpRequest: dtHRid
70   attribute volunteerStatus: etVolunteerStatus
71   operation is() : ptBoolean
72   operation init(AvId : dtInteger,Adisp : dtDispo,Aname : dtName, APhone : dtPhoneNumber,
         ACoordinates : dtCoordinates, AUsername : dtUserName,APassword :dtPassword) : ptBoolean
73  }
74
75  class ctHelpRequest role rnctHelpRequest cardinality [1..*] {
76   attribute HrTime: dtTime
77   attribute priority: dtInteger
78   attribute handlingWaitingTime: dtTime
79   attribute HelpRequestStatus: etHelpRequestStatus
80   operation is() : ptBoolean
81   operation init(AHrTime : dtTime) : ptBoolean
82
83  }
84
85  association assHelpRequest ctHelpRequest(rnctHelpRequest)[1..1] ctWeakPerson(rnctWeakPerson)[0..*]
86  association assCoordinatorHelpRequest ctCoordinator(rnctCoordinator)[0..1] ctHelpRequest(rnctHR)
        [0..*]
87  association assActCoordinatorVolunteer actVolunteer(roleOfC1forC2)[1..1] actAuthPerson(
        roleOfC2forC1)[0..0]
88  association assCheckBWeakPerson ctCheckBox(roleOfC1forC2)[0..1] ctHelpRequest(roleOfC2forC1)[1..1]
89  association assVolunteerHR ctVolunteer(roleOfC1forC2)[0..1] ctHelpRequest(roleOfC2forC1)[0..1]
90
91 }
92 }
93 }
```

Listing B.12: Messir Spec. file primarytypes-classes.msr.

B.13 File ./src-gen/messir-spec/concepts/primarytypes-datatypes/primarytypes-datatypes.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.concepts.primarytypes.datatypes {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12
13 Concept Model {
14
15  Primary Types {
16     datatype dtPhoneNumber {
17      attribute value : ptInteger
18      operation is() : ptBoolean
19     }
20
21     datatype dtDispo {
22      attribute time : dtDateAndTime
23     }
24     datatype dtHRid {
25      attribute value : ptInteger
26      operation is() : ptBoolean
27     }
28
```

```
29     datatype dtName {
30      attribute value : ptString
31      operation is() : ptBoolean
32     }
33     datatype dtCoordinates {
34      attribute long : ptReal
35      attribute lat : ptReal
36      operation is() : ptBoolean
37     }
38
39     datatype dtAddress {
40      attribute num : ptInteger
41      attribute street : ptString
42      attribute city : ptString
43     }
44     datatype dtMessage {
45      attribute MessageText : ptString
46      operation is() : ptBoolean
47     }
48
49     datatype dtRange {
50      attribute value : ptInteger
51      operation is() : ptBoolean
52     }
53     datatype dtUserName {
54      attribute value : ptString
55      operation is() : ptBoolean
56     }
57
58     datatype dtPassword {
59      attribute value : ptString
60      operation is() : ptBoolean
61     }
62
63     enum etHelpRequestStatus {
64      constants["processed", "not processed","handled","not handled"]
65     }
66
67     enum etVolunteerStatus {
68      constants["online", "occupied","offline"]
69     }
70  }
71 }
72 }
```

Listing B.13: Messir Spec. file primarytypes-datatypes.msr.

B.14 File ./src-gen/messir-spec/concepts/secondarytypes-associations/secondarytypes-associations.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.concepts.secondarytypes.associations {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12
13 Concept Model {
14
15  Secondary Types {
16
17  }
18 }
```

```
19 }
```

Listing B.14: Messir Spec. file secondarytypes-associations.msr.

B.15 File ./src-gen/messir-spec/concepts/secondarytypes-classes/secondarytypes-classes.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.concepts.secondarytypes.classes {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12
13 Concept Model {
14
15  Secondary Types {
16
17  }
18 }
19 }
```

Listing B.15: Messir Spec. file secondarytypes-classes.msr.

B.16 File ./src-gen/messir-spec/concepts/secondarytypes-datatypes/secondarytypes-datatypes.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.concepts.secondarytypes.datatypes {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12
13 Concept Model {
14
15  Secondary Types {
16   datatype dtTime {
17    attribute value : ptString
18   }
19
20  }
21
22 }
23 }
```

Listing B.16: Messir Spec. file secondarytypes-datatypes.msr.

B.17 File ./src-gen/messir-spec/tests/tests.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
```

```
5
6 package lu.uni.lassy.excalibur.g01.specification.tests {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12
13  Test Model {
14
15  }
16
17 }
```

Listing B.17: Messir Spec. file tests.msr.

B.18 File ./src-gen/messir-spec/concepts/uc$_c$oordinator.msr

```
1 /*
2 * @author Adriano
3 * @date Wed Nov 16 16:19:01 CET 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.usecases.coordinator {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12 import lu.uni.lassy.excalibur.g01.specification.environment
13
14  Use Case Model {
15   use case instance uciGetPendingHelpRequests : ugRetrievePendingHelpRequestDetails() {
16    actors {
17     tom : actCoordinator
18
19    }
20    use case steps {
21     tom executed instanceof subfunction oeGetPendingHelpRequests() {
22       ieSendPendingHR("12h40","661234567") returned to tom
23       ieSendPendingHR("12h43","661334569") returned to tom
24       ieSendPendingHR("12h48","691007221") returned to tom
25
26      }
27
28    tom executed instanceof subfunction oeGetHelpRequestDetail("id : 32") {
29      ieSendHelpRequestDetail("Paul Kremmer" , "6611726354", "49.618416, 6.172678" , "1 rue de Cents
            - Luxembourg") returned to tom
30
31      }
32
33    }
34   }
35   use case instance uciCallSelectedHelp: suCallSelectedHelpRequest{
36    actors{
37     tom : actCoordinator
38     tango : actPhoneCompany
39    }
40    use case steps {
41     tom executed instanceof subfunction oeReqCall("661234567"){
42       ieConfirm("661234567") returned to tango
43      }
44     tango executed instanceof subfunction oeGetConfirm("661234567","true"){
45      ieRequestConfirm("661234567","true") returned to tom
46     }
47
48    }
49   }
50
```

```
51  use case instance uciAssignPriority: ugAssignPriorityToHelpRequest{
52   actors{
53    tom : actCoordinator
54   }
55   use case steps {
56    tom executed instanceof subfunction oeReqCheckbox(){
57      ieSendCheckbox("Speaks slowly","Needs Water","Conscious") returned to tom
58     }
59    tom executed instanceof subfunction oeSendFilledCheckbox("Yes","Yes","Yes"){
60     ieSendCalculatedPriority("The calculated priority is: Urgent") returned to tom
61    }
62
63    tom executed instanceof subfunction oeConfirmPriority("OK"){
64     ieSendResult("Urgent") returned to tom
65    }
66   }
67
68  }
69
70  use case system summary suCallSelectedHelpRequest() {
71
72   actor actCoordinator[primary,active]
73   actor actPhoneCompany[secondary]
74
75   step a : actCoordinator executes oeReqCall()
76   step b : actPhoneCompany executes oeGetConfirm()
77
78  }
79
80  use case system subfunction oeReqCall() {
81   actor actCoordinator[primary,active]
82
83   returned messages{
84    ieConfirm(dtPhoneNumber) returned to actCoordinator
85   }
86  }
87
88  use case system subfunction oeGetConfirm() {
89   actor actPhoneCompany[primary,active]
90
91   returned messages{
92    ieRequestConfirm(AdtPhoneNumber, dtBoolean) returned to actPhoneCompany
93   }
94  }
95 use case system usergoal ugRetrievePendingHelpRequestDetails() {
96  actor actCoordinator[primary,active]
97
98  step a : actCoordinator executes oeGetPendingHelpRequests()
99  step b : actCoordinator executes oeGetHelpRequestDetail()
100 }
101
102 use case system subfunction oeGetPendingHelpRequests() {
103  actor actCoordinator[primary,active]
104
105  returned messages{
106
107   ieSendPendingHR(AdtTime) returned to actCoordinator
108
109  }
110 }
111
112 use case system subfunction oeGetHelpRequestDetail() {
113  actor actCoordinator[primary,active]
114
115  returned messages {
116
117   ieSendHelpRequestDetail() returned to actCoordinator
118  }
119 }
120
```

```
121  use case system usergoal ugAssignPriorityToHelpRequest(){
122  actor actCoordinator [primary,active]
123
124  step a : actCoordinator executes oeReqCheckbox()
125  step b : actCoordinator executes oeSendFilledCheckbox()
126  step c : actCoordinator executes oeConfirmPriority()
127 }
128
129  use case system subfunction oeReqCheckbox(){
130   actor actCoordinator[primary,active]
131
132   returned messages{
133    ieSendCheckbox(AdtCheckbox) returned to actCoordinator
134   }
135
136  }
137
138  use case system subfunction oeSendFilledCheckbox(){
139   actor actCoordinator[primary,active]
140
141   returned messages{
142    ieSendCalculatedPriority(dtString) returned to actCoordinator
143   }
144
145  }
146
147  use case system subfunction oeConfirmPriority(){
148   actor actCoordinator[primary,active]
149
150   returned messages{
151    ieSendResult(dtString) returned to actCoordinator
152   }
153
154 }
155
156 }
157
158 }
```

Listing B.18: Messir Spec. file uc$_c$*oordinator.msr*.

B.19 File ./src-gen/messir-spec/concepts/uc$_v$*olunteer.msr*

```
1 /*
2 * @author Adriano
3 * @date Sat Nov 19 15:04:07 CET 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.usecases.volunteer {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12 import lu.uni.lassy.excalibur.g01.specification.environment
13 import lu.uni.lassy.excalibur.g01.specification.usescases.weakperson
14
15  Use Case Model {
16   use case instance uciGetInRangeMission : ugGetMissionInRange {
17    actors {
18     MikeGPS : actPositionRequester
19     Mike : actVolunteer
20     Garmin : actSensor
21    }
22    use case steps {
23     MikeGPS executed instanceof subfunction oeGetPositionFromSensor() {
24       iereqCurrentPos() returned to Garmin
25      }
26     Garmin executed instanceof subfunction oeSendPos("49.626506, 6.163065") {
```

```
27       ieSentCurrentPosition("49.626506, 6.163065") returned to Mike
28      }
29     Mike executed instanceof subfunction oeGetInRangeMission("25","49.63999, 6.16399") {
30      ieSendInRangeMission("id=29","49.626506, 6.163065") returned to Mike
31      ieSendInRangeMission("id=34","49.627499, 6.274156") returned to Mike
32      ieSendInRangeMission("id=18","49.639006, 6.263164") returned to Mike
33
34      }
35    }
36   }
37
38   use case instance uciGetCurrentPositon : ugGetCurrentPosition {
39    actors {
40     Mike : actVolunteer
41     Garmin : actSensor
42    }
43    use case steps {
44     Mike executed instanceof subfunction oeGetCurrentPosition() {
45       ieRequestPos() returned to Garmin
46      }
47     Garmin executed instanceof subfunction oeSendCurrentPosition() {
48       ieReturnCurentPosition("49.626506, 6.163065") returned to Mike
49      }
50      Mike executed instanceof subfunction oeRegisterPosition("49.626506, 6.163065") {
51
52      }
53
54    }
55   }
56
57   use case system usergoal ugGetMissionInRange() {
58    actor actVolunteer[primary, active]
59    actor actPositionRequester[secondary]
60    actor actSensor[secondary]
61
62    step a: actPositionRequester executes oeGetPositionFromSensor()
63    step b: actSensor executes oeSendPos()
64    step c: actVolunteer executes oeGetInRangeMission()
65   }
66
67   use case system usergoal ugGetCurrentPosition() {
68    actor actVolunteer[primary,active]
69    actor actSensor[secondary]
70
71    step a : actVolunteer executes oeGetCurrentPosition()
72    step b : actSensor executes oeSendCurrentPosition()
73    step c : actVolunteer executes oeRegisterPosition()
74   }
75   use case system subfunction oeGetCurrentPosition() {
76    actor actVolunteer[primary,active]
77
78    returned messages{
79     ieRequestPos() returned to actSensor
80    }
81
82   }
83
84   use case system subfunction oeSendCurrentPosition() {
85    actor actSensor[primary,active]
86
87    returned messages{
88     ieReturnCurentPosition(adtCoordinates) returned to actVolunteer
89    }
90   }
91   use case system subfunction oeGetInRangeMission(){
92    actor actVolunteer[primary,active]
93
94    returned messages{
95     ieSendInRangeMission() returned to actVolunteer
96    }
```

```
97   }
98
99   use case system summary suRetrieveMissionDetails(){
100   actor actVolunteer[primary,active]
101
102   step a: actVolunteer executes ugGetMissionInRange()
103   step b: actVolunteer executes oeGetMissionDetails()
104  }
105
106  use case system subfunction oeGetMissionDetails(){
107   actor actVolunteer[primary, active]
108
109   returned messages{
110    ieSendMissionDetails() returned to actVolunteer
111   }
112  }
113
114  use case system summary suAcceptMission(){
115   actor actVolunteer[primary,active]
116
117   step a: actVolunteer executes ugGetMissionInRange()
118   step b: actVolunteer executes oeAcceptMission()
119  }
120
121  use case system subfunction oeAcceptMission() {
122   actor actVolunteer[primary, active]
123
124   returned messages{
125    ieSendDistanceMessage() returned to actVolunteer
126   }
127  }
128
129  use case system subfunction oeRegisterPosition() {
130   actor actVolunteer[primary,active]
131
132  }
133
134 }
135
136 }
```

Listing B.19: Messir Spec. file uc$_v$*olunteer.msr*.

B.20 File ./src-gen/messir-spec/usecases/usecases.msr

```
1 /*
2 * @author Adriano
3 * @date Sat Oct 22 12:57:25 CEST 2016
4 */
5
6 package lu.uni.lassy.excalibur.g01.specification.usescases.weakperson.old {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.math
10 import lu.uni.lassy.messir.libraries.primitives
11 import lu.uni.lassy.messir.libraries.string
12 import lu.uni.lassy.excalibur.g01.specification.environment
13
14  Use Case Model {
15
16  }
17 }
```

Listing B.20: Messir Spec. file usecases.msr.

B.21 File ./src-gen/messir-spec/usecases/weakperson.msr

```
1 /*
```

```
 * @author Carlos
 * @date Wed Nov 30 12:33:17 CET 2016
 */

package lu.uni.lassy.excalibur.g01.specification.usescases.weakperson {

import lu.uni.lassy.messir.libraries.calendar
import lu.uni.lassy.messir.libraries.math
import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.string
import lu.uni.lassy.excalibur.g01.specification.environment

 Use Case Model {

  use case instance uciSendHelpRequest : ugRequestHelp {
   actors {

    Benny : actWeakPerson
    BennyGPS : actPositionRequester
    Garmin : actSensor
    Tango : actPhoneCompany

   }
   use case steps {
    BennyGPS executed instanceof subfunction oeGetPositionFromSensor() {
      iereqCurrentPos() returned to Garmin
     }
    Garmin executed instanceof subfunction oeSendPos("49.626506, 6.163065") {
      ieSentdPosition("49.626506, 6.163065") returned to Benny
     }
    Benny executed instanceof subfunction oeInputManualPos("1 rue des mara chers-Luxembourg", "
        49.626506, 6.163065") {
      ieConfirmPos("1 rue des mara chers-Luxembourg", "49.626506, 6.163065") returned to Benny
     }
    Benny executed instanceof subfunction oeSendHelpRequest("Benny Hill" , "6611726354", "49.618416,
         6.172678" , "1 rue de Cents - Luxembourg") {
      ieSentConfirmationMessage("Your help request has been sent") returned to Benny
     }

    Benny executed instanceof subfunction oeAlertFamily("Hid : 12") {
       ieSmsForFamily("Benny has sent an help request","6611726354") returned to Tango
     }

   }
  }

  use case system summary ugRequestHelp() {

   actor actWeakPerson[primary,active]
   actor actSensor[secondary]
   actor actPositionRequester[secondary]

   reuse oeSendHelpRequest[1..1]

   step a: actPositionRequester executes oeGetPositionFromSensor()
   step b : actSensor executes oeSendPos()
   step c : actWeakPerson executes oeInputManualPos()
   step d : actWeakPerson executes oeSendHelpRequest()
   step e : actWeakPerson executes oeAlertFamily()

   ordering constraint "step (a) is always the first step "
   ordering constraint "step (b) must be executed after step (a) "
   ordering constraint "step (e) is always the last step "

  }

  use case system subfunction oeGetPositionFromSensor() {
   actor actPositionRequester[primary,active]
   actor actSensor[secondary]

```

```
70    returned messages{
71     iereqCurrentPos() returned to actSensor
72    }
73   }
74
75   use case system subfunction oeSendPos() {
76    actor actSensor[primary,active]
77    actor actWeakPerson[secondary]
78    actor actVolunteer[secondary]
79
80    returned messages {
81     ieSentdPosition() returned to actWeakPerson
82     ieSentCurrentPosition() returned to actVolunteer
83    }
84   }
85
86   use case system subfunction oeInputManualPos() {
87    actor actWeakPerson[primary,active]
88
89    returned messages{
90     ieConfirmPos() returned to actWeakPerson
91    }
92   }
93
94   use case system subfunction oeSendHelpRequest() {
95    actor actWeakPerson[primary,active]
96
97    returned messages{
98     ieSentConfirmationMessage() returned to actWeakPerson
99    }
100  }
101
102  use case system summary suAltertAFamilyMember() {
103   actor actWeakPerson [primary, active]
104   actor actPhoneCompany[secondary, active]
105
106   reuse oeAlertFamily[1..*]
107   reuse oeSendDeliveryReport[1..*]
108
109   step a: actWeakPerson executes oeAlertFamily()
110   step b: actPhoneCompany executes oeSendDeliveryReport()
111
112  }
113
114  use case system subfunction oeAlertFamily() {
115   actor actWeakPerson[primary,active]
116
117   returned messages {
118    ieSmsForFamily() returned to actPhoneCompany
119   }
120  }
121
122  use case system subfunction oeSendDeliveryReport() {
123   actor actPhoneCompany[primary,active]
124
125   returned messages{
126    ieMessageReception() returned to actPhoneCompany
127
128   }
129  }
130
131  use case instance uciAlertTheFamily : suAltertAFamilyMember() {
132   actors {
133    John : actWeakPerson
134    Tango : actPhoneCompany
135
136   }
137   use case steps {
138    John executed instanceof subfunction oeAlertFamily() {
139
```

```
140      ieSmsForFamily("John has just used the request help function through the Heat Wave Prevention
             System, serious danger may exist for this person", "691 432 132") returned to Tango
141      }
142    Tango executed instanceof subfunction oeSendDeliveryReport() {
143     ieMessageReception() returned to Tango
144      }
145
146    }
147   }
148
149  }
150 }
```

Listing B.21: Messir Spec. file weakperson.msr.

Bibliography

[1] Guelfi, N.: Messir: A Scientific Method for the Software Engineer. to be published (2017)

[2] Armour, F., Miller, G.: Advanced Use Case Modeling: Software Systems. Addison-Wesley (2001)