Peter the Great St.Petersburg Polytechnic University
Russia, 195251, St.Petersburg, Polytechnicheskaya, 29
Ksenia Panova

MyProjectName : Your Title
Messir Analysis Document
- v 0.0 -

(*Report type: Default*)

Thursday 10th November, 2016 - 05:39

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-suGlobalCrisisHandling

Figure 2.1 the actCoordinator's goal is to monitor the alerts received and the corresponding crisis in order to act as necessary to handle the crisis. Also to manage a list of media.

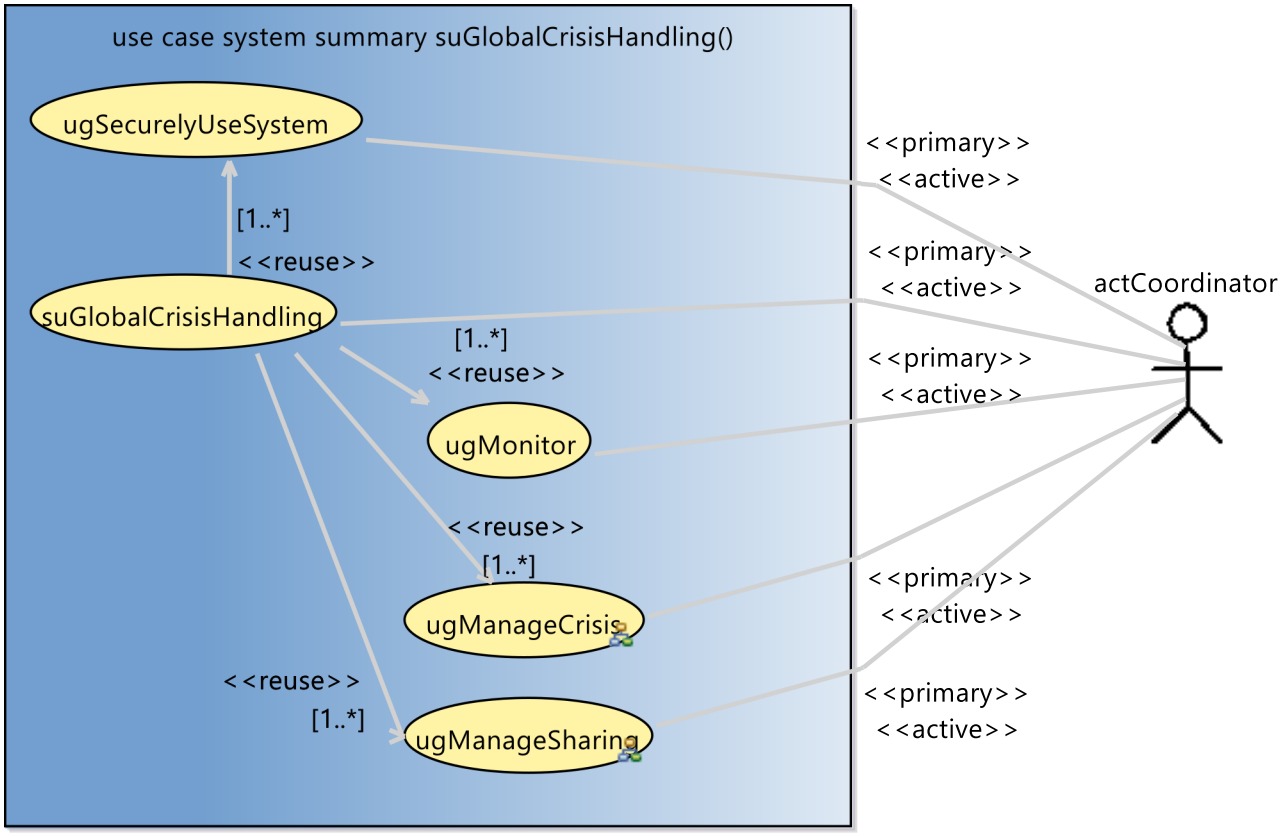


Figure 2.1:

2.3.1.2 usergoal-ugManageCrisis

Figure 2.2 The actCoordinator's goal is to get the detailed list of existing crisis or alerts to decide on next actions to undertake, for example to determine whether they are suitable to be sent to special media and send them.

2.3.1.3 usergoal-ugManageSharing

Figure 2.3 actCoordinator's goal consists in management of the list of media, namely, addition and removal of specialized media in the table

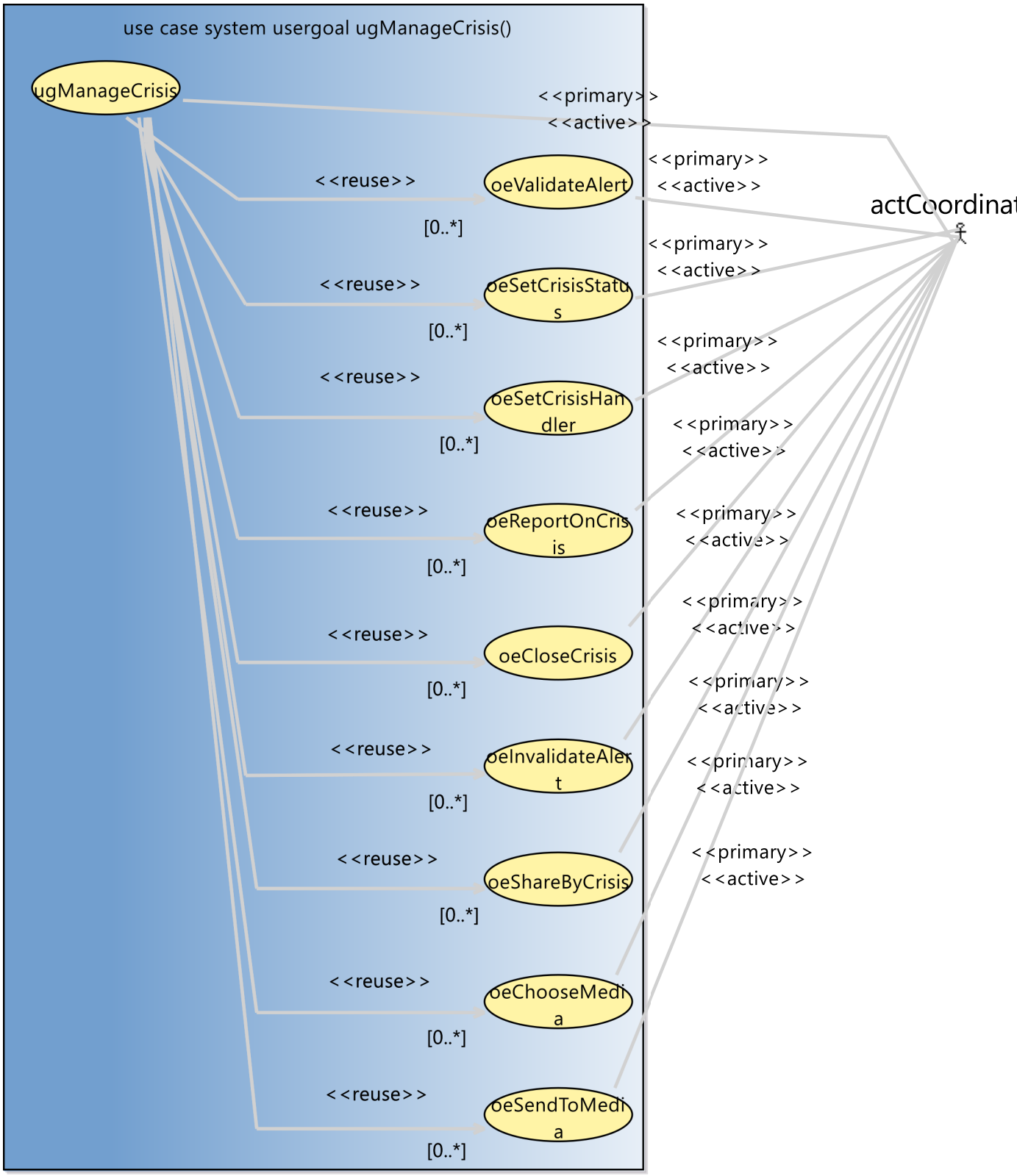

use case system usergoal ugManageCrisis()
ugManageCrisis
<<primary>>
<<active>>
<<reuse>>
oeValidateAlert
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeSetCrisisStatus
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeSetCrisisHandler
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeReportOnCrisis
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeCloseCrisis
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeInvalidateAlert
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeShareByCrisis
[0..*]
<<primary>>
<<active>>
<<reuse>>
oeChooseMedia
[0..*]
<<reuse>>
oeSendToMedia
[0..*]
actCoordinat

Figure 2.2:

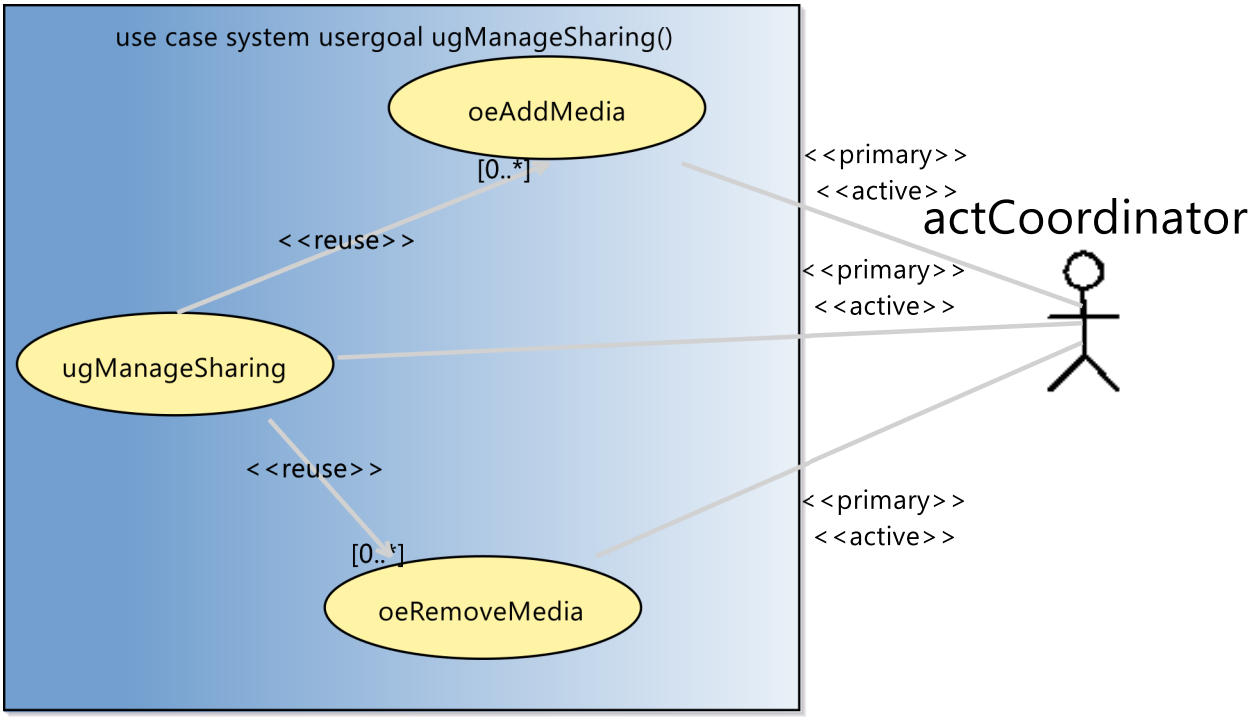


Figure 2.3:

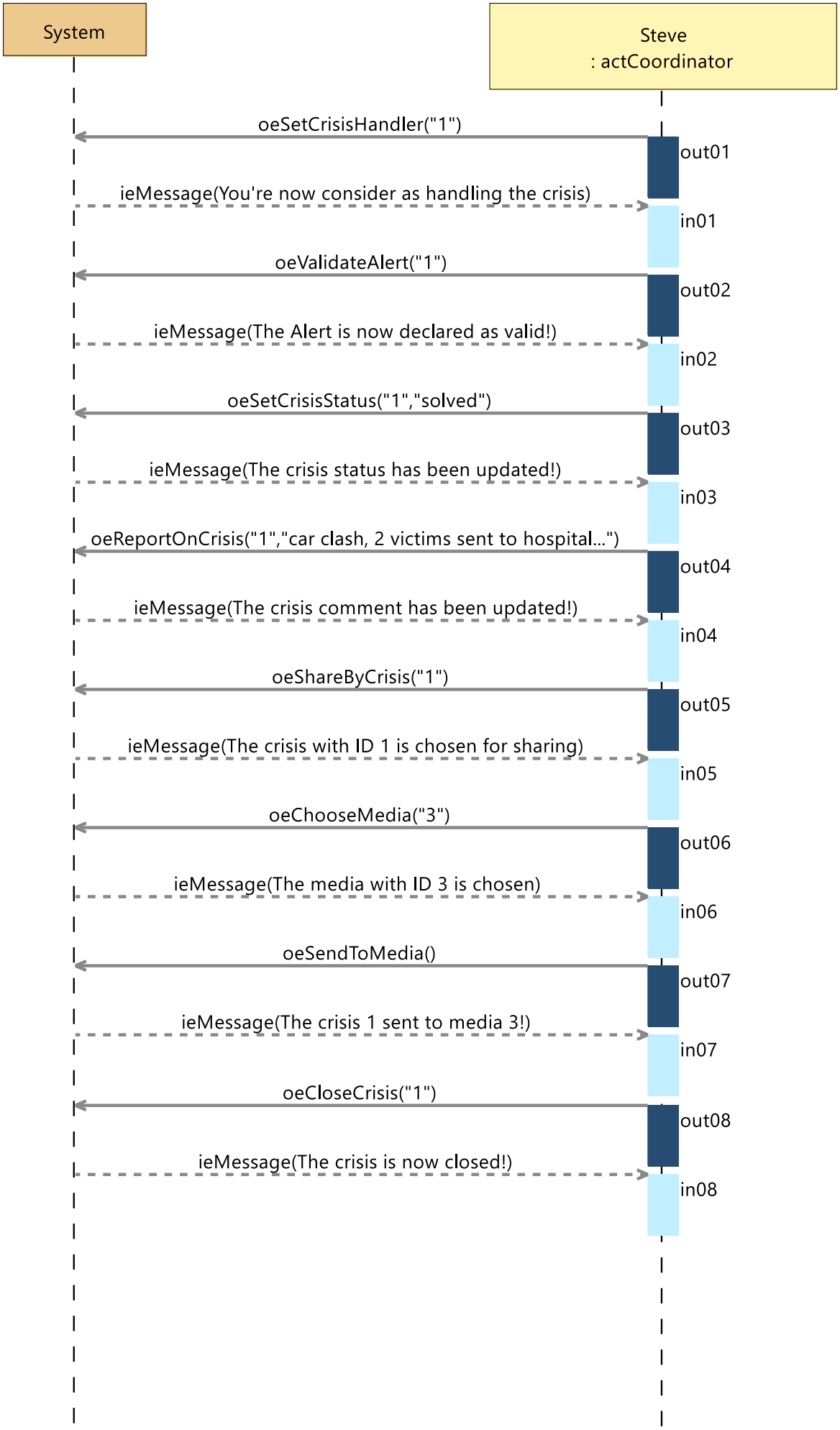
2.3.2 Use Case Instance(s)

2.3.2.1 Use-Case Instance - uciugManageCrisis:ugManageCrisis

Figure 2.4 Simple use case instance for the user goal `ugManageCrisis` illustrating a simple interaction scenario primarily handled by an coordinator in a concrete situation.Begins after the Coordinator got set of crises and when he sets crisis handler. he logs in the system until the full handling of all the existing crisis. After handling of crisis Coordinator determines a category of crisis and marks it for sharing with specific media

2.3.2.2 Use-Case Instance - uciugManageSharing:ugManageSharing

Figure 2.5 use case instance for the user goal `ugManageSharing` illustrates a simple example of scenario handled by an coordinator for management of the list of media in a concrete situation.


System
Steve
: actCoordinator
oeSetCrisisHandler("1")
out01
ieMessage(You're now consider as handling the crisis)
in01
oeValidateAlert("1")
out02
ieMessage(The Alert is now declared as valid!)
in02
oeSetCrisisStatus("1","solved")
out03
ieMessage(The crisis status has been updated!)
in03
oeReportOnCrisis("1","car clash, 2 victims sent to hospital...")
out04
ieMessage(The crisis comment has been updated!)
in04
oeShareByCrisis("1")
out05
ieMessage(The crisis with ID 1 is chosen for sharing)
in05
oeChooseMedia("3")
out06
ieMessage(The media with ID 3 is chosen)
in06
oeSendToMedia()
out07
ieMessage(The crisis 1 sent to media 3!)
in07
oeCloseCrisis("1")
out08
ieMessage(The crisis is now closed!)
in08

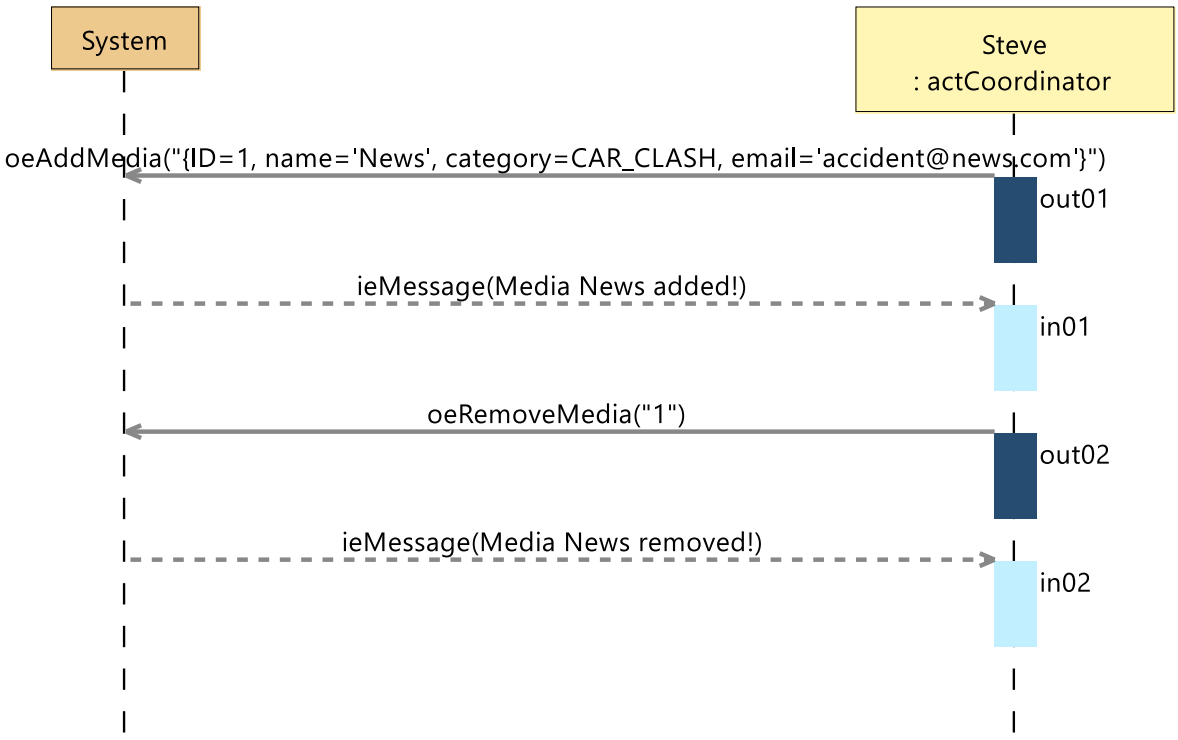

System
Steve
: actCoordinator
oeAddMedia("{ID=1, name='News', category=CAR_CLASH, email='accident@news.com'}")
out01
ieMessage(Media News added!)
in01
oeRemoveMedia("1")
out02
ieMessage(Media News removed!)
in02

Figure 2.5:

Chapter 3

Environment Model

3.1 Environment model view(s)

There are no view(s) for the Messir environment model.

3.2 Actors and Interfaces Descriptions

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

3.2.1 `actCoordinator` Actor

Actor	
actCoordinator	
represents actor responsible of handling one or several crisis for the *iCrash* system.	
Extends	
lu.uni.lassy.excalibur.icrash.spec.environment.actAuthenticated	
OutputInterfaces	
OUT 1	**`oeInvalidateAlert(AdtAlertID:dtAlertID):ptBoolean`**
	sent to indicate that an alert should be considered as closed.
OUT 2	**`oeCloseCrisis(AdtCrisisID:dtCrisisID):ptBoolean`**
	sent to indicate that a crisis should be considered as closed.
OUT 3	**`oeGetAlertsSet(AetAlertStatus:etAlertStatus):ptBoolean`**
	sent to request all the ctAlert instances having a specific status.
OUT 4	**`oeGetCrisisSet(AetCrisisStatus:etCrisisStatus):ptBoolean`**
	sent to request all the ctCrisis instances having a specific status.
OUT 5	**`oeSetCrisisHandler(AdtCrisisID:dtCrisisID):ptBoolean`**
	sent to declare himself as been the handler of a crisis having the specified id.
OUT 6	**`oeReportOnCrisis(AdtCrisisID:dtCrisisID, AdtComment:dtComment):ptBoolean`**
	sent to update the textual information available for a specific handled crisis.
OUT 7	**`oeSetCrisisStatus(AdtCrisisID:dtCrisisID, AetCrisisStatus:etCrisisStatus`**
	sent to define the handling status of a specific crisis.
OUT 8	**`oeSetCrisisType(AdtCrisisID:dtCrisisID, AetCrisisType:etCrisisType):ptBo`**
	sent to define the gravity type of a specific crisis.
OUT 9	**`oeValidateAlert(AdtAlertID:dtAlertID):ptBoolean`**

continues in next page . . .

... Actor table continuation

	sent to indicate that a specific alert is not a fake.
OUT 10	**oeAddMedia(ActMedia:ctMedia):ptBoolean**
	sent to indicate that a new media were added.
OUT 11	**oeRemoveMedia(AdtMediaID:dtMediaID):ptBoolean**
	sent to indicate that specific media were removed.
OUT 12	**oeShareByCrisis(AdtCrisisID:dtCrisisID):ptBoolean**
	sent to indicate that crises were chosen to share.
OUT 13	**oeChooseMedia(AdtMediaID:dtMediaID):ptBoolean**
	sent to indicate that specific media were chosen to share.
OUT 14	**oeSendToMedia():ptBoolean**
	sent to indicate that crises descriptions were sent to media.
InputInterfaces	
IN 1	**ieSendAnAlert(ActAlert:ctAlert):ptBoolean**
	allows for receiving a requested ctAlert instance.
IN 2	**ieSendACrisis(ActCrisis:ctCrisis):ptBoolean**
	allows for receiving a requested ctCrisis instance.

Chapter 4

Concept Model

4.1 PrimaryTypes-Classes

4.1.1 Local view 04

Figure 4.1 shows the local view of the ctMedia primary class type.

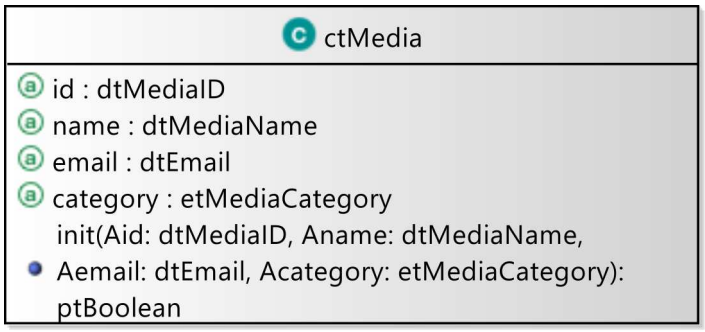


Figure 4.1: Concept Model - PrimaryTypes-Classes local view 04. local view of the ctMedia primary class type.

4.2 PrimaryTypes-Datatypes

4.2.1 Local view 05

Figure 4.2 used to determine which category of media waits specific crisis description

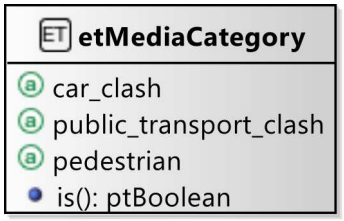


Figure 4.2: Concept Model - PrimaryTypes-Datatypes local view 05. .

4.3 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.

4.3.1 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.

Classes	
ctMedia	
determine instance of specific media	
attribute	**category: etMediaCategory**
attribute	**email: dtEmail**
attribute	**id: dtMediaID**
attribute	**name: dtMediaName**
operation	**init(Aid:dtMediaID, Aname:dtMediaName, Aemail:dtEmail, Acategory:etMediaCategory):ptBoolean**

4.3.2 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	
dtEmail	
primary data type for email of specific media	
extends	dtString
operation	**is():ptBoolean**
dtMediaID	
primary data type for media id	
extends	dtString
operation	**is():ptBoolean**
dtMediaName	
primary data type for media name	
extends	dtString
operation	**is():ptBoolean**

Enumerations

continues in next page . . .

...Enumerations table continuation

etMediaCategory used to determine which category of media waits specific crisis description.
operation **is():ptBoolean**

4.3.3 Primary types - Association types descriptions

There are no association types for the primary types.

4.3.4 Primary types - Aggregation types descriptions

There are no aggregation types for the primary types.

4.3.4.1 Primary types - Composition types descriptions

There are no composition types for the primary types.

4.3.5 Secondary types - Class types descriptions

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

4.3.6 Secondary types - Datatypes types descriptions

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

4.3.7 Secondary types - Association types descriptions

There are no association types for the secondary types.

4.3.8 Secondary types - Aggregation types descriptions

There are no aggregation types for the secondary types.

4.3.9 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 actCoordinator

5.1.1 Operation Model for oeAddMedia

The `oeAddMedia` operation has the following properties:

OPERATION	
oeAddMedia adding to common list of media	
Parameters	
1	**ActMedia: ctMedia** instance of media for adding to common list
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the system is started
PreP 2	the actor logged previously and did not log out ! (i.e. the associated ctCoordinator instance is considered logged)
Pre-Condition (functional)	
PreF 1	none
Post-Condition (functional)	
PostF 1	actCoordinator is informed about the finished adding operation
Post-Condition (protocol)	
PostP 1	none

5.1.2 Operation Model for oeChooseMedia

The `oeChooseMedia` operation has the following properties:

Operation	
oeChooseMedia	
selection of the specific media from the list to share with them	
Parameters	
1	**AdtMediaID: dtMediaID** chosen media id
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the system is started
PreP 2	the actor logged previously and did not log out ! (i.e. the associated ctCoordinator instance is considered logged)
Pre-Condition (functional)	
PreF 1	it is supposed that there exist at least one of the media with the appropriate category in the list
PreF 2	it assumes that you have selected the appropriate crisis
Post-Condition (functional)	
PostF 1	actCoordinator is informed about the choosing media
Post-Condition (protocol)	
PostP 1	none

5.1.3 Operation Model for oeCloseCrisis

The `oeCloseCrisis` operation has the following properties:

Operation	
oeCloseCrisis	
sent to indicate that a crisis should be considered as closed.	
Parameters	
1	**AdtCrisisID: dtCrisisID** the identification information used to determine the crisis to close
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the system is started
PreP 2	the actor logged previously and did not log out ! (i.e. the associated ctCoordinator instance is considered logged)
Pre-Condition (functional)	
PreF 1	it is supposed that there exist one ctCrisis instance with the same `id` attribute value as the one provided by the coordinator actor who wants to close.
Post-Condition (functional)	
PostF 1	the ctCrisis class instance having the provided id is considered closed in the post state.
PostF 2	There is no handler declared in the system as associated to the crisis.
PostF 3	all the alert instances associated to this crisis do not belong any more to the system's post state.
PostF 4	the coordinator actor is informed about the satisfaction of its request.
Post-Condition (protocol)	

continues in next page . . .

. . . Operation table continuation

PostP 1	none

5.1.4 Operation Model for oeRemoveMedia

The `oeRemoveMedia` operation has the following properties:

Operation

oeRemoveMedia	
removing media from the common list	
Parameters	
1	**AdtMediaID: dtMediaID**
	remove the media from passed media id
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the system is started
PreP 2	the actor logged previously and did not log out ! (i.e. the associated ctCoordinator instance is considered logged)
Pre-Condition (functional)	
PreF 1	it is supposed that there exist at least one media in the list
Post-Condition (functional)	
PostF 1	actCoordinator is informed about the finished removing operation
Post-Condition (protocol)	
PostP 1	none

5.1.5 Operation Model for oeSendToMedia

The `oeSendToMedia` operation has the following properties:

Operation

oeSendToMedia	
sent information about chosen crises to selected media	
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the system is started
PreP 2	the actor logged previously and did not log out ! (i.e. the associated ctCoordinator instance is considered logged)
Pre-Condition (functional)	
PreF 1	it is supposed that there exist at least one of the media with the appropriate category in the list
PreF 2	it assumes that have already been selected crises and the corresponding media
Post-Condition (functional)	
PostF 1	actCoordinator is informed about the sending information about crises
Post-Condition (protocol)	
PostP 1	none

5.1.6 Operation Model for oeShareByCrisis

The `oeShareByCrisis` operation has the following properties:

Operation

oeShareByCrisis	
sent to indicate that crisis was chosen to share.	
Parameters	
1	**AdtCrisisID: dtCrisisID** marked crisis id (may list)
Return type	
ptBoolean	
Pre-Condition (protocol)	
PreP 1	the system is started
PreP 2	the actor logged previously and did not log out ! (i.e. the associated ctCoordinator instance is considered logged)
Pre-Condition (functional)	
PreF 1	it is supposed that there exist at least one of the media with the appropriate category in the list
Post-Condition (functional)	
PostF 1	actCoordinator is informed about the chosen crises for sharing
Post-Condition (protocol)	
PostP 1	none

5.2 Environment - Actor Operation Schemes

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

5.3 Primary Types - Operation Schemes for Classes

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

5.4 Primary Types - Operation Schemes for Datatypes

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

5.5 Primary Types - Operation Schemes for Enumerations

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

5.6 Secondary Types - Operation Schemes for Classes

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

5.7 Secondary Types - Operation Schemes for Datatypes

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

5.8 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.icrash.spec.usecases.suGlobalCrisisHandling

A.1.2 Undocumented User-Goal Level Use Cases

- lu.uni.lassy.excalibur.icrash.spec.usecases.ugManageCrisis
- lu.uni.lassy.excalibur.icrash.spec.usecases.ugManageSharing
- lu.uni.lassy.excalibur.icrash.spec.usecases.ugMonitor
- lu.uni.lassy.excalibur.icrash.spec.usecases.ugSecurelyUseSystem

A.1.3 Undocumented Subfunction Level Use Cases

- lu.uni.lassy.excalibur.icrash.spec.usecases.oeAddMedia
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeChooseMedia
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeCloseCrisis
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeGetAlertsSet
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeGetCrisisSet
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeInvalidateAlert
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeRemoveMedia
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeReportOnCrisis
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeSendToMedia
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeSetCrisisHandler
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeSetCrisisStatus

- lu.uni.lassy.excalibur.icrash.spec.usecases.oeShareByCrisis
- lu.uni.lassy.excalibur.icrash.spec.usecases.oeValidateAlert

A.2 Undocumented Use Case Instances

A.2.1 Undocumented User-Goal Level Use Case Instances

- usecases.uciugManageSharing.uciugManageSharing
- usecases.uciugManageCrisis.uciugManageCrisis

A.3 Undocumented Actors

- lu.uni.lassy.excalibur.icrash.spec.environment.actActivator
- lu.uni.lassy.excalibur.icrash.spec.environment.actAdministrator
- lu.uni.lassy.excalibur.icrash.spec.environment.actAuthenticated
- lu.uni.lassy.excalibur.icrash.spec.environment.actComCompany
- lu.uni.lassy.excalibur.icrash.spec.environment.actMsrCreator

A.4 Undocumented Primary Types

A.4.1 Undocumented Primary Classe Types

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctAdministrator
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctAlert
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctAuthenticated
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctCoordinator
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctCrisis
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctHuman
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctState

A.4.2 Undocumented Primary Datatype Types

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtAlertID
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtComment
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtCoordinatorID
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtCrisisID
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtGPSLocation
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtLatitude

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtLogin
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtLongitude
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtPassword
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtPhoneNumber

A.4.3 Undocumented Primary Enumeration Types

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etAlertStatus
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etCrisisStatus
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etCrisisType
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etHumanKind

A.5 Undocumented Primary Relationships

A.5.1 Undocumented Primary Type Associations

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.associations.assctAlertctCrisis
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.associations.assctAlertctHuman
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.associations.assctAuthenticatedactAuthenticated
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.associations.assctCoordinatoractCoordinator
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.associations.assctCrisisctCoordinator
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.associations.assctHumanactComCompany

A.6 Undocumented Secondary Types

A.6.1 Undocumented Secondary Datatype Types

- lu.uni.lassy.excalibur.icrash.spec.concepts.secondarytypes.datatypes.dtSMS

A.7 Undocumented Operation Specifications

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctAlert.isSentToCoordinator
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctCrisis.handlingDelayPassed
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctCrisis.isAllocatedIfPossible
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctCrisis.isSentToCoordinator
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctCrisis.maxHandlingDelayPassed
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes.ctHuman.isAcknowledged
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtAlertID.is

- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtComment.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtCoordinatorID.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtCrisisID.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtEmail.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtGPSLocation.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtGPSLocation.isNearTo
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtLatitude.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtLogin.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtLongitude.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtMediaID.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtMediaName.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtPassword.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.dtPhoneNumber.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etAlertStatus.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etCrisisStatus.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etCrisisType.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etHumanKind.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes.etMediaCategory.is
- lu.uni.lassy.excalibur.icrash.spec.concepts.secondarytypes.datatypes.dtSMS.is
- lu.uni.lassy.excalibur.icrash.spec.environment.actActivator.outactActivator.oeSetClock
- lu.uni.lassy.excalibur.icrash.spec.environment.actActivator.outactActivator.oeSollicitateCrisisHandling
- lu.uni.lassy.excalibur.icrash.spec.environment.actAdministrator.outactAdministrator.oeAddCoordinator
- lu.uni.lassy.excalibur.icrash.spec.environment.actAdministrator.outactAdministrator.oeDeleteCoordinat
- lu.uni.lassy.excalibur.icrash.spec.environment.actAuthenticated.outactAuthenticated.oeLogin
- lu.uni.lassy.excalibur.icrash.spec.environment.actAuthenticated.outactAuthenticated.oeLogout
- lu.uni.lassy.excalibur.icrash.spec.environment.actComCompany.outactComCompany.oeAlert
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeGetAlertsSet
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeGetCrisisSet
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeInvalidateAlert
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeReportOnCrisis

- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeSetCrisisHandler
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeSetCrisisStatus
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeSetCrisisType
- lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.oeValidateAlert
- lu.uni.lassy.excalibur.icrash.spec.environment.actMsrCreator.outactMsrCreator.oeCreateSystemAndEnvironm

Appendix B

Messir Specification Files Listing

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

```
//
//DON'T TOUCH THIS FILE !!!
//
package uuid6972196d7b104531a20b45e09943cd53 {
 Concept Model {}
}
```

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

B.2 File ./src-gen/messir-spec/operations/environment/environment-actCoordinator-oeAddMedia.msr

```
package lu.uni.lassy.excalibur.icrash.spec.environment.operations.actCoordinator.outactCoordinator.
    oeAddMedia {

import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.math
import lu.uni.lassy.messir.libraries.string
import lu.uni.lassy.messir.libraries.calendar
import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes

 Operation Model {

  operation: lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.
      oeAddMedia(ActMedia:ctMedia):ptBoolean{
  // include below the specification information (pre,post or ocl or prolog)

  }
 }
}
```

Listing B.2: Messir Spec. file environment-actCoordinator-oeAddMedia.msr.

B.3 File ./src-gen/messir-spec/operations/environment/environment-actCoordinator-oeChooseMedia.msr

```
package lu.uni.lassy.excalibur.icrash.spec.environment.operations.actCoordinator.outactCoordinator.
    oeChooseMedia {

import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.math
import lu.uni.lassy.messir.libraries.string
import lu.uni.lassy.messir.libraries.calendar
import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes
```

```
8
9  Operation Model {
10
11   operation: lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.
         oeChooseMedia(AdtMediaID:dtMediaID):ptBoolean{
12   // include below the specification information (pre,post or ocl or prolog)
13
14   }
15  }
16 }
```

Listing B.3: Messir Spec. file environment-actCoordinator-oeChooseMedia.msr.

B.4 File ./src-gen/messir-spec/operations/environment/environment-actCoordinator-oeCloseCrisis.msr

```
1 package lu.uni.lassy.excalibur.icrash.spec.environment.operations.actCoordinator.outactCoordinator.
      oeCloseCrisis {
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.icrash.spec.concepts.primarytypes.datatypes
8
9  Operation Model {
10
11   operation: lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.
         oeCloseCrisis(AdtCrisisID:dtCrisisID):ptBoolean{
12   // include below the specification information (pre,post or ocl or prolog)
13
14   }
15  }
16 }
```

Listing B.4: Messir Spec. file environment-actCoordinator-oeCloseCrisis.msr.

B.5 File ./src-gen/messir-spec/operations/environment/environment-actCoordinator-oeRemoveMedia.msr

```
1 package lu.uni.lassy.excalibur.icrash.spec.environment.operations.actCoordinator.outactCoordinator.
      oeRemoveMedia {
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.icrash.spec.concepts.primarytypes.datatypes
8
9  Operation Model {
10
11   operation: lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.
         oeRemoveMedia(AdtMediaID:dtMediaID):ptBoolean{
12   // include below the specification information (pre,post or ocl or prolog)
13
14   }
15  }
16 }
```

Listing B.5: Messir Spec. file environment-actCoordinator-oeRemoveMedia.msr.

B.6 File ./src-gen/messir-spec/operations/environment/environment-actCoordinator-oeSendToMedia.msr

```
1 package lu.uni.lassy.excalibur.icrash.spec.environment.operations.actCoordinator.outactCoordinator.
      oeSendToMedia {
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   operation: lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.
         oeSendToMedia():ptBoolean{
11   // include below the specification information (pre,post or ocl or prolog)
12
13   }
14  }
15 }
```

Listing B.6: Messir Spec. file environment-actCoordinator-oeSendToMedia.msr.

B.7 File ./src-gen/messir-spec/operations/environment/environment-actCoordinator-oeShareByCrisis.msr

```
1 package lu.uni.lassy.excalibur.icrash.spec.environment.operations.actCoordinator.outactCoordinator.
      oeShareByCrisis {
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.icrash.spec.concepts.primarytypes.datatypes
8
9  Operation Model {
10
11   operation: lu.uni.lassy.excalibur.icrash.spec.environment.actCoordinator.outactCoordinator.
         oeShareByCrisis(AdtCrisisID:dtCrisisID):ptBoolean{
12   // include below the specification information (pre,post or ocl or prolog)
13
14   }
15  }
16 }
```

Listing B.7: Messir Spec. file environment-actCoordinator-oeShareByCrisis.msr.

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

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:39 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.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.icrash.spec.concepts.primarytypes.associations
13 import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes
14 import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes
15 import lu.uni.lassy.excalibur.icrash.spec.concepts.secondarytypes.datatypes
16
17 Environment Model {
18
19   actor actMsrCreator role rnactMsrCreator cardinality [1..1] {
20
21    operation init():ptBoolean
```

```
22
23   input interface inactMsrCreator {
24   }
25   output interface outactMsrCreator {
26    operation oeCreateSystemAndEnvironment(AqtyComCompanies:ptInteger ):ptBoolean
27   }
28  }
29
30  actor actAdministrator
31      role rnactAdministrator
32      cardinality [1..1]
33      extends actAuthenticated {
34
35   operation init():ptBoolean
36
37   output interface outactAdministrator{
38
39    operation oeAddCoordinator(
40           AdtCoordinatorID:dtCoordinatorID ,
41           AdtLogin:dtLogin ,
42           AdtPassword:dtPassword ):ptBoolean
43
44    operation oeDeleteCoordinator(
45           AdtCoordinatorID:dtCoordinatorID ):ptBoolean
46   }
47
48   input interface inactAdministrator{
49
50    operation ieCoordinatorAdded():ptBoolean
51    operation ieCoordinatorDeleted():ptBoolean
52    }
53  }
54
55  actor actCoordinator
56      role rnactCoordinator
57      cardinality [0..*]
58      extends actAuthenticated{
59
60   operation init():ptBoolean
61
62   output interface outactCoordinator{
63    operation oeInvalidateAlert(AdtAlertID:dtAlertID ):ptBoolean
64    operation oeCloseCrisis(AdtCrisisID:dtCrisisID ):ptBoolean
65    operation oeGetAlertsSet(AetAlertStatus:etAlertStatus ):ptBoolean
66    operation oeGetCrisisSet(AetCrisisStatus:etCrisisStatus ):ptBoolean
67    operation oeSetCrisisHandler(AdtCrisisID:dtCrisisID ):ptBoolean
68    operation oeReportOnCrisis(
69           AdtCrisisID:dtCrisisID ,
70           AdtComment:dtComment
71           ):ptBoolean
72    operation oeSetCrisisStatus(
73           AdtCrisisID:dtCrisisID ,
74           AetCrisisStatus:etCrisisStatus
75           ):ptBoolean
76    operation oeSetCrisisType(
77                    AdtCrisisID:dtCrisisID ,
78                    AetCrisisType:etCrisisType
79                    ):ptBoolean
80    operation oeValidateAlert(AdtAlertID:dtAlertID ):ptBoolean
81
82    operation oeAddMedia(ActMedia:ctMedia):ptBoolean
83    operation oeRemoveMedia(AdtMediaID:dtMediaID):ptBoolean
84    operation oeShareByCrisis(AdtCrisisID:dtCrisisID):ptBoolean
85    operation oeChooseMedia(AdtMediaID:dtMediaID):ptBoolean
86    operation oeSendToMedia():ptBoolean
87   }
88
89   input interface inactCoordinator{
90    operation ieSendAnAlert(ActAlert:ctAlert ):ptBoolean
91    operation ieSendACrisis(ActCrisis:ctCrisis ):ptBoolean
```

```
92      }
93    }
94
95    actor actComCompany role rnactComCompany cardinality [0..*]{
96
97     operation init():ptBoolean
98
99     output interface outactComCompany{
100     operation oeAlert(
101           AetHumanKind:etHumanKind ,
102           AdtDate:dtDate ,
103           AdtTime:dtTime ,
104           AdtPhoneNumber:dtPhoneNumber ,
105           AdtGPSLocation:dtGPSLocation ,
106           AdtComment:dtComment
107           ):ptBoolean
108    }
109
110    input interface inactComCompany{
111     operation ieSmsSend(AdtPhoneNumber:dtPhoneNumber ,
112               AdtSMS:dtSMS
113               ):ptBoolean
114     }
115   }
116
117   actor actAuthenticated role rnactAuthenticated cardinality [0..*]{
118
119    operation init():ptBoolean
120
121    output interface outactAuthenticated{
122     operation oeLogin(AdtLogin:dtLogin , AdtPassword:dtPassword ):ptBoolean
123     operation oeLogout():ptBoolean
124    }
125
126    input interface inactAuthenticated{
127     operation ieMessage(AMessage:ptString):ptBoolean
128     }
129   }
130
131   actor actActivator[proactive] role rnactActivator cardinality [1..1]{
132
133    operation init():ptBoolean
134
135    output interface outactActivator{
136     proactive operation oeSollicitateCrisisHandling():ptBoolean
137     proactive operation oeSetClock(AcurrentClock:dtDateAndTime ):ptBoolean
138    }
139
140    input interface inactActivator{
141     }
142   }
143  }
144 }
```

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

B.9 File ./src-gen/messir-spec/concepts/primarytypes-associations/primarytypes-associations.msr

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:38 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.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 import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes
13 import lu.uni.lassy.excalibur.icrash.spec.environment
14
15  Concept Model {
16
17   Primary Types{
18
19 // Internal
20
21 association assctAlertctCrisis
22  ctAlert(rnAlerts)[1..*]
23  ctCrisis (rnTheCrisis)[1..1]
24
25 association assctAlertctHuman
26  ctAlert(rnSignaled)[1..*]
27  ctHuman (rnSignaler)[1..1]
28
29 association assctCrisisctCoordinator
30  ctCrisis(rnHandled)[0..*]
31  ctCoordinator(rnHandler)[0..1]
32
33 // With Actors
34
35    association assctHumanactComCompany
36       ctHuman(rnctHuman)[0..*]
37       actComCompany(rnactComCompany)[1..1]
38
39    association assctCoordinatoractCoordinator
40       ctCoordinator(rnctCoordinator)[1..1]
41       actCoordinator(rnactCoordinator)[1..1]
42
43    association assctAuthenticatedactAuthenticated
44       ctAuthenticated(rnctAuthenticated)[1..1]
45       actAuthenticated(rnactAuthenticated)[1..1]
46
47   }
48  }
49 }
```

Listing B.9: Messir Spec. file primarytypes-associations.msr.

B.10 File ./src-gen/messir-spec/concepts/primarytypes-classes/primarytypes-classes.msr

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:38 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.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.icrash.spec.concepts.primarytypes.associations
13 import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes
14 import lu.uni.lassy.excalibur.icrash.spec.concepts.secondarytypes.datatypes
15 import lu.uni.lassy.excalibur.icrash.spec.environment
16
17  Concept Model {
18
19   Primary Types{
20
21    state class ctState {
22     attribute nextValueForAlertID:dtInteger
```

```
23    attribute nextValueForCrisisID:dtInteger
24    attribute clock:dtDateAndTime
25    attribute crisisReminderPeriod:dtSecond
26    attribute maxCrisisReminderPeriod:dtSecond
27    attribute vpLastReminder:dtDateAndTime
28    attribute vpStarted:ptBoolean
29
30    operation init( AnextValueForAlertID:dtInteger,
31            AnextValueForCrisisID:dtInteger,
32             Aclock:dtDateAndTime,
33            AcrisisReminderPeriod:dtSecond ,
34            AmaxCrisisReminderPeriod:dtSecond ,
35            AvpLastReminder:dtDateAndTime ,
36            AvpStarted:ptBoolean ): ptBoolean
37  }
38
39  class ctAlert role rnctAlert cardinality [0..*]{
40    attribute id:dtAlertID
41    attribute status: etAlertStatus
42    attribute location:dtGPSLocation
43    attribute instant:dtDateAndTime
44    attribute comment:dtComment
45
46    operation init(    Aid:dtAlertID ,
47          Astatus:etAlertStatus ,
48          Alocation:dtGPSLocation ,
49          Ainstant:dtDateAndTime ,
50          Acomment:dtComment ):ptBoolean
51    operation isSentToCoordinator(AactCoordinator:actCoordinator ):ptBoolean
52
53  }
54
55  class ctCrisis role rnctCrisis cardinality [0..*]{
56    attribute id:dtCrisisID
57    attribute type:etCrisisType
58    attribute status: etCrisisStatus
59    attribute location:dtGPSLocation
60    attribute instant:dtDateAndTime
61    attribute comment:dtComment
62
63    operation init(
64          Aid:dtCrisisID ,
65          Atype:etCrisisType ,
66          Astatus:etCrisisStatus ,
67          Alocation:dtGPSLocation ,
68          Ainstant:dtDateAndTime ,
69          Acomment:dtComment ):ptBoolean
70
71    operation handlingDelayPassed():ptBoolean
72    operation maxHandlingDelayPassed():ptBoolean
73    operation isSentToCoordinator(AactCoordinator:actCoordinator ):ptBoolean
74    operation isAllocatedIfPossible():ptBoolean
75  }
76
77  class ctHuman role rnctHuman cardinality [0..*]{
78    attribute id:dtPhoneNumber
79    attribute kind:etHumanKind
80
81    operation init(
82          Aid:dtPhoneNumber ,
83          Akind:etHumanKind ):ptBoolean
84    operation isAcknowledged():ptBoolean
85  }
86
87  class ctAuthenticated
88     role rnctAuthenticated
89     cardinality [0..*]{
90
91    attribute login:dtLogin
92    attribute pwd: dtPassword
```

```
93     attribute vpIsLogged:ptBoolean
94
95     operation init(
96           Alogin:dtLogin ,
97           Apwd:dtPassword ):ptBoolean
98    }
99
100   class ctCoordinator
101     role rnctCoordinator
102     cardinality [0..*]
103     extends ctAuthenticated{
104
105    attribute id:dtCoordinatorID
106
107    operation init(
108          Aid:dtCoordinatorID ,
109          Alogin:dtLogin ,
110          Apwd:dtPassword ):ptBoolean
111   }
112
113   class ctAdministrator
114     role rnctAdministrator
115     cardinality [1..1]
116     extends ctAuthenticated{
117
118    operation init(
119          Alogin:dtLogin ,
120          Apwd:dtPassword ):ptBoolean
121   }
122
123   class ctMedia
124     role rnctMedia
125     cardinality [0..*] {
126
127    attribute id:dtMediaID
128    attribute name:dtMediaName
129    attribute email:dtEmail
130    attribute category:etMediaCategory
131
132    operation init(
133          Aid:dtMediaID ,
134          Aname:dtMediaName ,
135          Aemail:dtEmail ,
136          Acategory:etMediaCategory):ptBoolean
137
138   }
139  }
140 }
141 }
```

Listing B.10: Messir Spec. file primarytypes-classes.msr.

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

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:38 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.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
17   datatype dtAlertID extends dtString {
18    operation is():ptBoolean
19   }
20   datatype dtCrisisID extends dtString {
21    operation is():ptBoolean
22   }
23   datatype dtLogin extends dtString {
24    operation is():ptBoolean
25   }
26   datatype dtPassword extends dtString {
27    operation is():ptBoolean
28   }
29   datatype dtCoordinatorID extends dtString {
30    operation is():ptBoolean
31   }
32   datatype dtPhoneNumber extends dtString {
33    operation is():ptBoolean
34   }
35   datatype dtComment extends dtString {
36    operation is():ptBoolean
37   }
38   datatype dtLatitude extends dtReal {
39    operation is():ptBoolean
40   }
41   datatype dtLongitude extends dtReal {
42    operation is():ptBoolean
43   }
44   datatype dtGPSLocation {
45    attribute latitude: dtLatitude
46    attribute longitude: dtLongitude
47    operation is():ptBoolean
48    operation isNearTo(AGPSLocation:dtGPSLocation ):ptBoolean
49   }
50
51   datatype dtMediaID extends dtString {
52    operation is():ptBoolean
53   }
54   datatype dtMediaName extends dtString {
55    operation is():ptBoolean
56   }
57   datatype dtEmail extends dtString {
58    operation is():ptBoolean
59   }
60
61   enum etCrisisStatus {
62    constants["pending", "handled", "solved","closed"]
63    operation is():ptBoolean
64   }
65   enum etAlertStatus {
66    constants["pending", "valid", "invalid"]
67    operation is():ptBoolean
68   }
69   enum etCrisisType {
70    constants["small", "medium", "huge"]
71    operation is():ptBoolean
72   }
73   enum etHumanKind {
74    constants["witness", "victim", "anonymous"]
75    operation is():ptBoolean
76   }
77   enum etMediaCategory{
78    constants["car_clash", "public_transport_clash", "pedestrian"]
79    operation is():ptBoolean
80   }
81  }
82 }
```

```
83 }
```

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

B.12 File ./src-gen/messir-spec/concepts/secondarytypes-associations/secondarytypes-associations.msr

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:38 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.concepts.secondarytypes.associations {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.primitives
10 import lu.uni.lassy.messir.libraries.math
11 import lu.uni.lassy.messir.libraries.string
12
13 Concept Model {
14
15  Secondary Types {
16
17  }
18 }
19 }
```

Listing B.12: Messir Spec. file secondarytypes-associations.msr.

B.13 File ./src-gen/messir-spec/concepts/secondarytypes-classes/secondarytypes-classes.msr

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:38 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.concepts.secondarytypes.classes {
7
8 import lu.uni.lassy.messir.libraries.calendar
9 import lu.uni.lassy.messir.libraries.string
10 import lu.uni.lassy.messir.libraries.math
11 import lu.uni.lassy.messir.libraries.primitives
12
13 Concept Model {
14
15  Secondary Types {
16
17  }
18 }
19 }
```

Listing B.13: Messir Spec. file secondarytypes-classes.msr.

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

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:39 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.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 dtSMS {
17     attribute value: ptString
18     operation is():ptBoolean
19    }
20  }
21
22 }
23 }
```

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

B.15 File ./src-gen/messir-spec/tests/tests.msr

```
 1 /*
 2 * @author Kseniya
 3 * @date Wed Nov 09 14:31:39 MSK 2016
 4 */
 5
 6 package lu.uni.lassy.excalibur.icrash.spec.tests {
 7
 8 import lu.uni.lassy.messir.libraries.calendar
 9 import lu.uni.lassy.messir.libraries.string
10 import lu.uni.lassy.messir.libraries.math
11 import lu.uni.lassy.messir.libraries.primitives
12
13  Test Model {
14
15  }
16
17 }
```

Listing B.15: Messir Spec. file tests.msr.

B.16 File ./src-gen/messir-spec/usecases/usecaseinstance-ugManageCrisis-uciugManageCrisis.msr

```
 1 package usecases.uciugManageCrisis {
 2  import lu.uni.lassy.excalibur.icrash.spec.usecases
 3  import lu.uni.lassy.excalibur.icrash.spec.environment
 4  import lu.uni.lassy.excalibur.icrash.spec.usecases
 5
 6  Use Case Model {
 7
 8   use case instance uciugManageCrisis : ugManageCrisis{
 9    actors {
10     Steve:actCoordinator
11    }
12    use case steps {
13     Steve executed instanceof subfunction oeSetCrisisHandler("1"){
14      ieMessage("You're now consider as handling the crisis") returned to Steve
15     }
16
17     Steve executed instanceof subfunction oeValidateAlert("1"){
18      ieMessage("The Alert is now declared as valid!") returned to Steve
19     }
20
21     Steve executed instanceof subfunction oeSetCrisisStatus("1", "solved") {
22      ieMessage("The crisis status has been updated!") returned to Steve
```

```
23     }
24
25     Steve executed instanceof subfunction oeReportOnCrisis("1", "car clash, 2 victims sent to
           hospital...") {
26      ieMessage("The crisis comment has been updated!") returned to Steve
27     }
28
29     Steve executed instanceof subfunction oeShareByCrisis("1") {
30      ieMessage("The crisis with ID 1 is chosen for sharing") returned to Steve
31     }
32
33     Steve executed instanceof subfunction oeChooseMedia("3") {
34      ieMessage("The media with ID 3 is chosen") returned to Steve
35     }
36
37     Steve executed instanceof subfunction oeSendToMedia() {
38      ieMessage("The crisis 1 sent to media 3!") returned to Steve
39     }
40
41     Steve executed instanceof subfunction oeCloseCrisis("1") {
42      ieMessage("The crisis is now closed!") returned to Steve
43     }
44
45    }
46   }
47  }
48 }
```

Listing B.16: Messir Spec. file usecaseinstance-ugManageCrisis-uciugManageCrisis.msr.

B.17 File ./src-gen/messir-spec/usecases/usecaseinstance-ugManageSharing-uciugManageSharing.msr

```
1 package usecases.uciugManageSharing {
2  import lu.uni.lassy.excalibur.icrash.spec.usecases
3  import lu.uni.lassy.excalibur.icrash.spec.environment
4
5  Use Case Model {
6
7   use case instance uciugManageSharing : ugManageSharing {
8    actors {
9     Steve : actCoordinator
10   }
11   use case steps {
12    Steve executed instanceof subfunction oeAddMedia("{ID=1, name='News', category=CAR_CLASH, email
          ='accident@news.com'}") {
13     ieMessage("Media News added!") returned to Steve
14    }
15    Steve executed instanceof subfunction oeRemoveMedia("1") {
16     ieMessage("Media News removed!") returned to Steve
17    }
18   }
19  }
20 }
21 }
```

Listing B.17: Messir Spec. file usecaseinstance-ugManageSharing-uciugManageSharing.msr.

B.18 File ./src-gen/messir-spec/usecases/usecases.msr

```
1 /*
2 * @author Kseniya
3 * @date Wed Nov 09 14:31:38 MSK 2016
4 */
5
6 package lu.uni.lassy.excalibur.icrash.spec.usecases {
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.icrash.spec.environment
13 import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.classes
14 import lu.uni.lassy.excalibur.icrash.spec.concepts.primarytypes.datatypes
15
16  Use Case Model {
17
18 //-------------------------------------------------------
19    use case system subfunction oeAddMedia(ActMedia:ctMedia) {
20     actor actCoordinator[primary,active]
21     returned messages {
22      ieMessage(AMessage) returned to actCoordinator
23     }
24    }
25
26 //-------------------------------------------------------
27    use case system subfunction oeRemoveMedia(AdtMediaID:dtMediaID) {
28     actor actCoordinator[primary,active]
29     returned messages {
30      ieMessage(AMessage) returned to actCoordinator
31     }
32    }
33
34 //-------------------------------------------------------
35    use case system subfunction oeShareByCrisis(AdtCrisisID:dtCrisisID) {
36     actor actCoordinator[primary,active]
37     returned messages {
38      ieMessage(AMessage) returned to actCoordinator
39     }
40    }
41
42 //-------------------------------------------------------
43    use case system subfunction oeChooseMedia(AdtMediaID:dtMediaID) {
44     actor actCoordinator[primary,active]
45     returned messages {
46      ieMessage(AMessage) returned to actCoordinator
47     }
48    }
49
50 //-------------------------------------------------------
51    use case system subfunction oeSendToMedia() {
52     actor actCoordinator[primary,active]
53     returned messages {
54      ieMessage(AMessage) returned to actCoordinator
55     }
56    }
57
58 //-------------------------------------------------------
59 //-------------------------------------------------------
60 //-------------------------------------------------------
61
62   use case system usergoal ugManageSharing() {
63    actor actCoordinator[primary,active]
64
65    reuse oeAddMedia[0..*]
66    reuse oeRemoveMedia[0..*]
67
68    step a: actCoordinator executes oeAddMedia
69    step b: actCoordinator executes oeRemoveMedia
70
71   }
72
73 //-------------------------------------------------------
74 //-------------------------------------------------------
75 //-------------------------------------------------------
76
77 //-------------------------------------------------------
```

```
78  use case system subfunction oeValidateAlert(AdtAlertID:dtAlertID) {
79    actor actCoordinator[primary,active]
80    returned messages {
81     ieMessage(AMessage) returned to actCoordinator
82    }
83   }
84
85 //-----------------------------------------------------
86   use case system subfunction oeSetCrisisStatus(AdtCrisisID:dtCrisisID ,AetCrisisStatus:
         etCrisisStatus) {
87    actor actCoordinator[primary,active]
88    returned messages {
89     ieMessage(AMessage) returned to actCoordinator
90    }
91   }
92
93 //-----------------------------------------------------
94   use case system subfunction oeSetCrisisHandler(AdtCrisisID:dtCrisisID) {
95    actor actCoordinator[primary,active]
96    actor actCoordinator[secondary,passive]
97    actor actComCompany[secondary,passive,multiple]
98    returned messages {
99     ieMessage(AMessage)
100     returned to actCoordinator
101    ieSendAnAlert(ActAlert)
102     returned to actCoordinator
103    ieSmsSend(AdtPhoneNumber,AdtSMS)
104     returned to actComCompany
105   }
106  }
107
108 //-----------------------------------------------------
109  use case system subfunction oeCloseCrisis(AdtCrisisID:dtCrisisID) {
110   actor actCoordinator[primary,active]
111   returned messages {
112    ieMessage(AMessage) returned to actCoordinator
113   }
114 }
115
116 //-----------------------------------------------------
117  use case system subfunction oeReportOnCrisis(AdtCrisisID:dtCrisisID,AdtComment:dtComment) {
118   actor actCoordinator[primary,active]
119   returned messages {
120    ieMessage(AMessage) returned to actCoordinator
121   }
122  }
123
124 //-----------------------------------------------------
125  use case system subfunction oeInvalidateAlert(AdtAlertID:dtAlertID) {
126   actor actCoordinator[primary,active]
127   actor actComCompany[secondary,passive]
128   returned messages {
129    ieMessage(AMessage) returned to actCoordinator
130  }
131  }
132
133 //-----------------------------------------------------
134 //-----------------------------------------------------
135 //-----------------------------------------------------
136  use case system usergoal ugManageCrisis() {
137   actor actCoordinator[primary, active]
138
139   reuse oeValidateAlert[0..*]
140   reuse oeSetCrisisStatus[0..*]
141   reuse oeSetCrisisHandler[0..*]
142   reuse oeReportOnCrisis[0..*]
143   reuse oeCloseCrisis[0..*]
144   reuse oeInvalidateAlert[0..*]
145   reuse oeShareByCrisis[0..*]
146   reuse oeChooseMedia[0..*]
```

```
147   reuse oeSendToMedia[0..*]
148
149   step a: actCoordinator executes oeValidateAlert
150   step b: actCoordinator executes oeSetCrisisStatus
151   step c: actCoordinator executes oeSetCrisisHandler
152   step d: actCoordinator executes oeReportOnCrisis
153   step f: actCoordinator executes oeCloseCrisis
154   step g: actCoordinator executes oeInvalidateAlert
155   step h: actCoordinator executes oeShareByCrisis
156   step i: actCoordinator executes oeChooseMedia
157   step k: actCoordinator executes oeSendToMedia
158
159   ordering constraint "managing a crisis is doing one of the indicated use cases."
160  }
161
162 //-------------------------------------------------------
163 //-------------------------------------------------------
164 //-------------------------------------------------------
165
166 //-------------------------------------------------------
167   use case system subfunction oeGetAlertsSet(AetAlertStatus:etAlertStatus) {
168    actor actCoordinator[primary,active]
169    returned messages {
170     ieSendAnAlert(ActAlert) returned to actCoordinator
171    }
172   }
173 //-------------------------------------------------------
174   use case system subfunction oeGetCrisisSet(AetCrisisStatus:etCrisisStatus){
175    actor actCoordinator[primary,active]
176    returned messages {
177     ieSendACrisis(ActCrisis) returned to actCoordinator
178    }
179   }
180
181 //-------------------------------------------------------
182 //-------------------------------------------------------
183 //-------------------------------------------------------
184
185  use case system usergoal ugMonitor() {
186     actor actCoordinator[primary,active]
187
188    reuse oeGetCrisisSet[0..*]
189    reuse oeGetAlertsSet[0..*]
190
191    step a: actCoordinator executes oeGetAlertsSet
192    step b: actCoordinator executes oeGetCrisisSet
193   }
194
195 //-------------------------------------------------------
196 //-------------------------------------------------------
197 //-------------------------------------------------------
198
199   use case system usergoal ugSecurelyUseSystem() {
200     actor actCoordinator[primary,active]
201     // TODO
202    }
203
204 //-------------------------------------------------------
205 //-------------------------------------------------------
206 //-------------------------------------------------------
207
208  use case system summary suGlobalCrisisHandling() {
209   actor actCoordinator[primary,active]
210
211   reuse ugSecurelyUseSystem[1..*]
212   reuse ugMonitor[1..*]
213   reuse ugManageCrisis[1..*]
214   reuse ugManageSharing[1..*]
215
216   step a: actCoordinator executes ugSecurelyUseSystem
```

```
217     step b: actCoordinator executes ugMonitor
218     step c: actCoordinator executes ugManageCrisis
219     step d: actCoordinator executes ugManageSharing
220
221       ordering constraint "steps (a) (b) (c) and (d) executions are interleaved
222           (steps (b) (c) and (d) have their protocol constrained by steps of (a))."
223       ordering constraint "steps (a) (b) (c) and (d) can be executed multiple times."
224    }
225
226  }
227
228 }
```

Listing B.18: Messir Spec. file usecases.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)