Software use cases to elicit the software requirements analysis within the ASTRI project

Conforti, Vito, Antolini, Elisa, Bonnoli, Giacomo, Bruno, Pietro, Bulgarelli, Andrea, et al.

Vito Conforti, Elisa Antolini, Giacomo Bonnoli, Pietro Bruno, Andrea Bulgarelli, Milvia Capalbi, Valentina Fioretti, Dino Fugazza, Daniele Gardiol, Alessandro Grillo, Giuseppe Leto, Saverio Lombardi, Fabrizio Lucarelli, Maria Concetta Maccarone, Giuseppe Malaguti, Giovanni Pareschi, Federico Russo, Pierluca Sangiorgi, Joseph Schwarz, Salvatore Scuderi, Claudio Tanci, Gino Tosti, Massimo Trifoglio, Stefano Vercellone, Ricardo Zanmar Sanchez, "Software use cases to elicit the software requirements analysis within the ASTRI project," Proc. SPIE 9913, Software and Cyberinfrastructure for Astronomy IV, 991340 (26 July 2016); doi: 10.1117/12.2232292

Event: SPIE Astronomical Telescopes + Instrumentation, 2016, Edinburgh, United Kingdom

Software Use Cases to Elicit the Software Requirements Analysis within the ASTRI Project

Vito Conforti [*a], Elisa Antolini[b], Giacomo Bonnoli[c], Pietro Bruno[d], Andrea Bulgarelli [a], Milvia Capalbi [e], Valentina Fioretti [a], Dino Fugazza[c], Daniele Gardiol[f], Alessandro Grillo[d], Giuseppe Leto[d], Saverio Lombardi[g], Fabrizio Lucarelli[g], Maria Concetta Maccarone [e], Giuseppe Malaguti [a], Giovanni Pareschi[c], Federico Russo[f], Pierluca Sangiorgi [e], Joseph Schwarz[c], Salvatore Scuderi[d], Claudio Tanci[b], Gino Tosti[b], Massimo Trifoglio [a], Stefano Vercellone [e], Ricardo Zanmar Sanchez[d]
for the ASTRI Collaboration[h] and the CTA Consortium[i]

[a] INAF, IASF-Bologna, Via Pietro Gobetti 101, 40129 Bologna, Italy
[b]Università di Perugia, Via A. Pascoli, Perugia, Italy
[c]INAF – Osservatorio Astronomico di Brera, Via Bianchi 46, Merate, Italy
[d]INAF – Osservatorio Astronomico di Catania, Via S. Sofia 78, Catania, Italy
[e] INAF, IASF-Palermo, Via Ugo La Malfa 153, 90146 Palermo, Italy
[f]INAF – Osservatorio Astronomico di Torino, Via Osservatorio 20, Pino Torinese, Italy
[g]INAF – Osservatorio Astronomico di Roma, Via di Frascati 33, Monteporzio Catone (Roma), Italy
[h] http://www.brera.inaf.it/astri
[i]http://www.cta-observatory.org

ABSTRACT

The Italian National Institute for Astrophysics (INAF) is leading the Astrofisica con Specchi a Tecnologia Replicante Italiana (ASTRI) project whose main purpose is the realization of small size telescopes (SST) for the Cherenkov Telescope Array (CTA). The first goal of the ASTRI project has been the development and operation of an innovative end-to-end telescope prototype using a dual-mirror optical configuration (SST-2M) equipped with a camera based on silicon photo-multipliers and very fast read-out electronics. The ASTRI SST-2M prototype has been installed in Italy at the INAF "M.G. Fracastoro" Astronomical Station located at Serra La Nave, on Mount Etna, Sicily. This prototype will be used to test several mechanical, optical, control hardware and software solutions which will be used in the ASTRI mini-array, comprising nine telescopes proposed to be placed at the CTA southern site. The ASTRI mini-array is a collaborative and international effort led by INAF and carried out by Italy, Brazil and South-Africa. We present here the use cases, through UML (Unified Modeling Language) diagrams and text details, that describe the functional requirements of the software that will manage the ASTRI SST-2M prototype, and the lessons learned thanks to these activities. We intend to adopt the same approach for the Mini Array Software System that will manage the ASTRI mini-array operations. Use cases are of importance for the whole software life cycle; in particular they provide valuable support to the validation and verification activities. Following the iterative development approach, which breaks down the software development into smaller chunks, we have analysed the requirements, developed, and then tested the code in repeated cycles. The use case technique allowed us to formalize the problem through user stories that describe how the user procedurally interacts with the software system. Through the use cases we improved the communication among team members, fostered common agreement about system requirements, defined the normal and alternative course of events, understood better the business process, and defined the system test to ensure that the delivered software works properly. We present a summary of the ASTRI SST-2M prototype use cases, and how the lessons learned can be exploited for the ASTRI mini-array proposed for the CTA Observatory.

Keywords: Cherenkov Telescope Array, CTA, ASTRI, Software requirement, Use Cases

[*] conforti@iasfbo.inaf.it, http://www.iasfbo.inaf.it/~conforti/

Software and Cyberinfrastructure for Astronomy IV, edited by Gianluca Chiozzi,
Juan C. Guzman, Proc. of SPIE Vol. 9913, 991340 · © 2016 SPIE
CCC code: 0277-786X/16/$18 · doi: 10.1117/12.2232292

1. INTRODUCTION

The Italian National Institute for Astrophysics (INAF) is leading the ASTRI project[1] proposed for the ambitious Cherenkov Telescope Array (CTA)[2]. In the framework of the small size class of telescopes[3], a first goal of the ASTRI project is the realization of an end-to-end prototype in a dual-mirror configuration (SST-2M). The camera at the focal plane is composed of a matrix of Silicon photo-multiplier sensors managed by an innovative front-end and back-end electronics[4]. The ASTRI SST-2M prototype is installed in Italy at the INAF "M.G. Fracastoro" observing station located at Serra La Nave, 1735 m a.s.l. on Mount Etna, Sicily. As a second step, the ASTRI project is aiming to implement a mini-array[5] of CTA pre-production units, composed of nine ASTRI telescopes and proposed to be placed at the CTA southern site. ASTRI MASS[6] (Mini Array Software System) is the software system that oversees the telescope monitoring, control, data analysis and archive, as well as data access and dissemination at any level, including user or hardware. The MASS functional requirements are defined through the use cases presented in this paper. In Software and Systems Engineering, a use case[7] is a list of steps which describe a scenario, typically defining interactions between a role, known in Unified Modelling Language[8] (UML) as an "actor", and a system, to achieve a goal. The actor can be a human, an external system, or time. The use case, in compliance to the software requirements, details what the software shall do. We have defined the use cases in an iterative way taking into account the user requirements, the stakeholder goals, and the supplier experiences. The ASTRI MASS use cases are supporting the whole software lifecycle, and in particular the software architecture, design and code verifications. We present in this contribution the first version of the use cases to describe the user experience with the ASTRI SST-2M, which will be eventually refurbished to support the ASTRI mini-array. We are adopting the iterative method in order to define, in some iterations, the most real scenarios expected by the whole stakeholders according to the requirements. Section 2 of this paper, details the actors. Then, the section 3 provides, through the UML use case diagrams, an overview of the expected functionalities. In addition we report the details of some crucial use cases through text-form tables in section 3. Finally, the last section summarizes the next steps and lessons learned for the ASTRI SST-2M use cases.

2. THE ASTRI MASS ACTORS

The use case actors are all the people or software/hardware systems that interact with the system of interest to achieve the goal of the related use case.

We have defined the following human actors foreseen for the ASTRI SST-2M:

- Operator: who performs the telescope operations;
- Engineer: who performs assembly, integration, verification, test and maintenance operations;
- Astronomer on duty: who is the observation assistant;
- PI: who is the scientist responsible for a certain approved observing program;
- TAC (Time Allocation Committee): it is the team who sets the grade of the observing program;
- MC manager: who is the owner and creator of the Montecarlo simulation data;
- Pipeline manager: who is responsible for the science pipeline software execution.

3. USE CASES OVERVIEW

This section provides an overview of the ASTRI MASS use cases through the UML use case diagrams. The MASS stakeholders aim at a software system able to support the end-to-end telescope for the ASTRI project and which is compliant with the needs and requirements of the CTA observatory. Four main functionalities, sketched in figure 1, will be provided: schedule, control, monitor and analysis of the observation. Each use case mentioned in figure 1 is detailed in the following sub-sections.

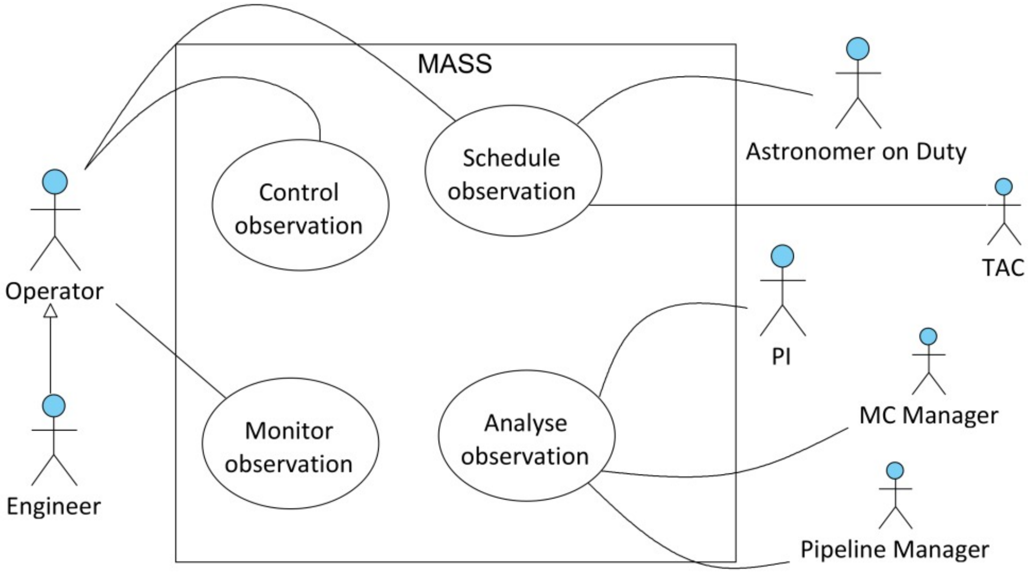

Figure 1. UML ASTRI MASS use case diagram.

3.1 Schedule the Observation

The MASS shall provide all functionalities, depicted in figure 2, for the observation scheduling. The Astronomer on Duty, following the PI instructions, submits the proposals that are ranked in order by the TAC people. Submission proposal and ranking activities may be performed also off-site. When the Operator begins the observing night on-site, they have to prepare the night schedule in order to define the scheduling blocks (SB) to be executed that night. Then the Operator begins the execution of the night observations that is to perform the sequence of the scheduled SBs. The scheduling blocks, which compound the proposal, are blocks of the observations that contain specific instrument configuration to observe a target following a defined sequence pattern.

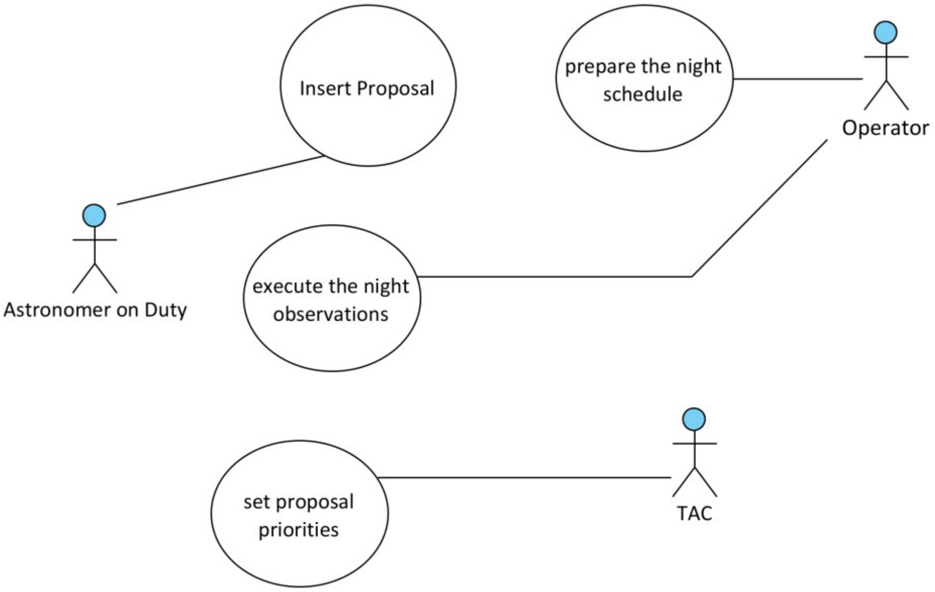

Figure 2. UML schedule the observation use case diagram.

3.2 Control Observation

Figure 3 represents the observation control scenarios. The Operator performs system start-up and technical pre-calibration, also during the daylight, before starting the operations. Once the observing night ends, the Operator performs the technical post-calibration and shut-down of the system. In case of system failure the Operator may need to reboot the system.

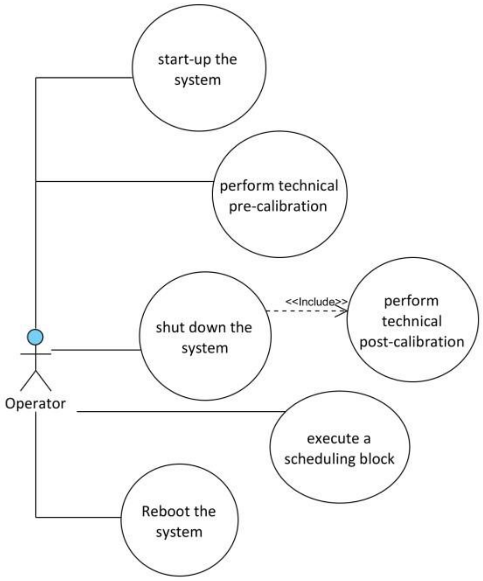
start-up the system
perform technical pre-calibration
shut down the system
<<Include>>
perform technical post-calibration
Operator
execute a scheduling block
Reboot the system

Figure 3. UML control the observation use case diagram.

3.3 Monitor Observation

The use case diagram in figure 4 depicts the monitoring scenarios. The continuous monitoring of the guard level ensures the operator that both the hardware and software systems are properly working. The "react to alarm" use case describes the system failure which requires human intervention. The Operator should always be able to write and read the operations log file. We foresee also specific system, weather and environmental condition monitoring that shall be compliant with the current scheduling block execution.

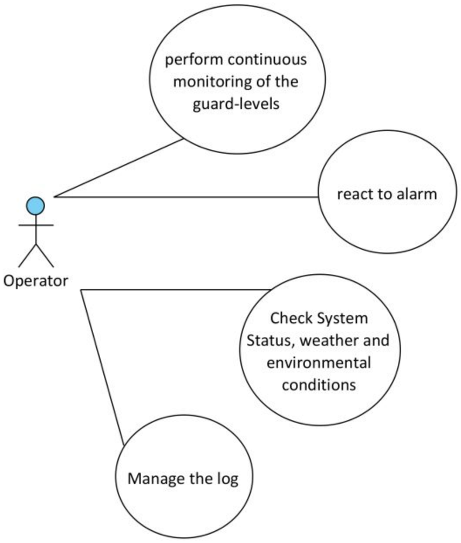
perform continuous monitoring of the guard-levels
react to alarm
Operator
Check System Status, weather and environmental conditions
Manage the log

Figure 4. UML monitor the observation use case diagram.

3.4 Analyse the Observation

The observation analysis collects many scenarios that we have grouped in three main use cases in figure 5. The MC Manager who is in charge of producing Monte Carlo simulation files to be archived. The Pipeline Manager analyses the data in many contexts and levels. The analysis data use case is detailed in other more specific use cases. For many actors such as the PI, the guest observer and the scientist, the final goal of the observation is to get data which is described through the proper scenario.

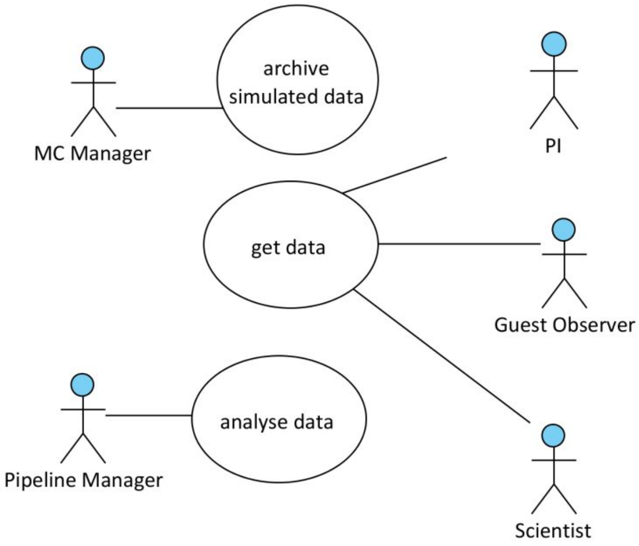

Figure 5. UML analyse the observation use case diagram.

4. USE CASE DETAILS

In this section we detail some of the most significant use cases among those presented in section 3. In the scenario description there are some MASS affected systems which are treated as a black box from the primary actor point of view. The list of the whole affected systems for the following use cases are: OCS (Observatory Control System), ICS (Instrument Control System), DAQ (Data AcQuisition system), DHS (Data Handling System), Scheduler, Auxiliary Control System, FOC (Fiber Optic Calibration system), PHP (Proposal Handling Platform).

Table 1. – Perform technical pre-calibrations use case detail.

Use Case	**Perform technical pre-calibrations**
Primary Actor	Operator
Pre-conditions	The instrument system, acquisition system and data handling system are ready. The telescope is in parking position. The camera lid is closed.
Post-conditions	The camera is calibrated, at least within acceptable limits. Data of all calibration runs are saved on disk; FOC is switched-off; the camera lid is closed.
Main success scenario	1. The operator through the OCS commands to perform the technical pre-calibrations 2. ICS, DAQ and DHS perform the dark stair calibration 3. ICS switch-on the FOC system 4. ICS, DAQ and DHS perform the fiber stair calibration 5. ICS, DAQ and DHS perform the fibre PEQ (Photoelectron equivalent) calibration 6. ICS switch-off the FOC system 7. ICS, DAQ and DHS perform the dark PEQ calibration
Extensions	2.a the dark stair execution fails

	2.a.1 ICS stops the calibrations 2.a.2 ICS notifies to OCS calibration failure to block the nightly observation plan 3.a the FOC does not switch-on 3.a.1 ICS notifies to OCS FOC failure and proceeds with step 7 of main scenario 4.a the fibre stair execution fails 4.a.1 ICS notifies to OCS calibration failure and proceeds with next step of main scenario 5.a the fibre PEQ execution fails 5.a.1 ICS notifies to OCS calibration failure and proceeds with next step of main scenario 6.a the FOC does not switch-off 6.a.1 ICS stop the calibration procedure 6.a.2 ICS notifies to OCS FOC failure to lock the nightly observation plan in order to ask a human technical intervention 7.a the dark PEQ execution fails 7.a.1 ICS stops the calibration procedure 7.a.2 ICS notifies to OCS this calibration failure (without stopping the nightly observation plan)

Table 2. – Execute a SB use case detail.

Use Case	**Execute a SB**
Primary Actor	OCS
Pre-conditions	The system is ready and the camera is calibrated, SB details are available
Post-conditions	The SB is executed and the SB status is notified
Main success scenario	1. OCS, through the scheduler, gets the suitable information of the next observable target within the night schedule 2. OCS reads and share to the TCS and ICS the SB details 3. TCS determines the target position 4. TCS slews the telescope to the target position 5. ICS configures the detectors 6. ICS configures the DAQ 7. OCS notify the Operator that the SB execution has been started 8. TCS starts the telescope tracking 9. ICS runs the exposure 10. DAQ acquires and displays data 11. DHS archives data 12. OCS commands the DHS to perform the scientific pipeline for real-time feedbacks and alerts 13. OCS notifies the Operator and the scheduler that the SB execution has been completed successfully
Extensions	9.a Auxiliary Control System detects the environment condition not compliant with the SB execution 9.a.1 OCS notifies to the Operator that the current environment conditions are not

	compliant with the SB execution 13.a the SB execution has been not completed successfully 13.a.1 the OCS notifies the Operator and the scheduler the SB is aborted

Table 3. – reboot the system use case detail.

Use Case	**Reboot the system**
Primary Actor	Engineer
Pre-conditions	The Scheduler, TCS, ICS, DAQ, Auxiliary Control System, DHS are ready or fault
Post-conditions	The Scheduler, TCS, ICS, DAQ, Auxiliary Control System, DHS are ready
Main success scenario	1. Engineer commands the OCS to reboot the system 2. OCS Shut down the ICS 3. OCS shut down the scheduler 4. OCS shut down the TCS 5. OCS shut down the DAQ 6. OCS shut down the Auxiliary Control System 7. OCS shut down the DHS 8. the engineer start-up the system
Extensions	2.a the ICS does not shut down 2.a.1 the OCS notifies the error to the engineer 2.a.2 the engineer hard shuts down the ICS 3.a the scheduler does not shut down 3.a.1 the OCS notifies the error to the engineer 3.a.2 the engineer hard shuts down the scheduler 4.a TCS does not shut down 4.a.1 OCS notifies the error to the engineer 4.a.2 engineer hard shuts down the TCS 5.a DAQ does not shut down 5.a.1 OCS notifies the error to the engineer 5.a.2 engineer hard shuts down the DAQ 6.a Auxiliary Control System does not shut down 6.a.1 OCS notifies the error to the engineer 6.a.2 engineer hard shuts down the Auxiliary Control System 7.a DHS does not shut down 7.a.1 OCS notifies the error to the engineer 7.a.2 engineer hard shuts down the DHS

Table 4. – Perform continuous monitoring use case detail.

Use Case	**Perform continuous monitoring**
Primary Actor	OCS

Pre-conditions	The Scheduler, TCS, ICS, DAQ, Auxiliary Control System, DHS are ready
Post-conditions	The OCS collects the monitoring status of the Scheduler, TCS, ICS, DAQ, Auxiliary Control System, DHS
Main success scenario	1. TCS, each predefined time, collects and sends to the OCS the monitoring data 2. ICS, each predefined time, collects and sends to the OCS the monitoring data 3. DAQ, each predefined time, collects and sends to the OCS the monitoring data 4. The Auxiliary Control System, each predefined time, collects and sends to the OCS the monitoring data 5. DHS, each predefined time, collects and sends to the OCS the monitoring data
Extensions	1.a the monitor data sending fails 1.a.1 TCS notifies the OCS that the monitoring fails 1.b the monitoring data denote a sub-system anomaly 1.b.1 the OCS notifies the anomaly to the Operator 2.a the monitor data sending fails 2.a.1 ICS notifies the OCS that the monitoring fails 2.b the monitoring data denote a sub-system anomaly 2.b.1 the OCS notifies the anomaly to the Operator 3.a the monitor data sending fails 3.a.1 DAQ notifies the OCS that the monitoring fails 3.b the monitoring data denote a sub-system anomaly 3.b.1 the OCS notifies the anomaly to the Operator 4.a the monitor data sending fails 4.a.1 the Auxiliary Control System notifies the OCS that the monitoring fails 4.b the monitoring data denote a sub-system anomaly 4.b.1 the OCS notifies the anomaly to the Operator 5.a the monitor data sending fails 5.a.1 DHS notifies the OCS that the monitoring fails 5.b the monitoring data denote a sub-system anomaly 5.b.1 the OCS notifies the anomaly to the Operator

Table 5. – check status, weather and environmental conditions use case detail.

Use Case	**Check status, weather and environmental conditions**
Primary Actor	Operator
Pre-conditions	Scheduler, TCS, ICS, DAQ, DHS, Auxiliary Control System are supporting the observation
Post-conditions	The information about system, weather and environment are collected and analysed
Main success	Continuously during the SB execution:

scenario	1. The OCS collects through ACS the system information; 2. The OCS analyse the system information 3. The OCS collects through the Auxiliary Control System the weather and environment condition data 4. The OCS analyses the weather and environment condition data
Extensions	2.a The health system condition is not suitable to support the SB execution 2.a.1 OCS aborts the SB execution 2.a.2 OCS notifies to the operator the system status 2.a.3 OCS notifies the scheduler the SB is aborted 4.a the weather or environment conditions are not adequate to grant the SB execution 4.a.1 interrupt the SB execution 4.a.2 notifies the operator the environment status 4.a.3 find the best SB compatible with the current environment status

Table 6. – Insert proposal use case detail.

Use Case	**Insert proposal**
Primary Actor	Astronomer on Duty
Pre-conditions	The PHP is ready
Post-conditions	The PHP is ready
Main success scenario	1. The Astronomer on Duty connects and authenticate the PHP 2. The PHP authorize the Astronomer on Duty access 3. The Astronomer on Duty compose and submit the proposal to the PHP 4. The PHP archives the proposal 5. The PHP notifies the Astronomer on Duty the operation is successfully completed
Extensions	2a Astronomer on Duty credentials are not valid 2a.1 the PHP displays the error; 2a.2 the PHP denies the access to the Astronomer on Duty; 3a the proposal is not completed 3a.1the PHP displays the error;

Table 7. – Execute night observation use case detail.

Use Case	**Execute night observations**
Primary Actor	Operator
Pre-conditions	The night schedule with SBs is available
Post-conditions	Night schedule is empty
Main success scenario	1. The operator commands the OCS to start the night schedule 2. For each SB in the night schedule

	1. The OCS gets the SB through the scheduler 2. The scheduler executes the SB 3. The scheduler notifies the operator the status of SB 3. The scheduler notifies the night schedule is completed
Extensions	

5. DISCUSSION AND CONCLUSIONS

We have presented in this contribution the use cases to describe the scenarios for the ASTRI SST-2M. These use cases will be used to validate the MASS software architecture and to support software integration and test activities. Since the MASS is a complex system of heterogeneous sub-systems, it was challenging to collect the stakeholder expectancy. In particular to meet the customer needs (science and management teams) and the knowledge suppliers (sub-system engineer teams). We solved this issue through many interactions among the stakeholders and by distributing a clear use case template which has driven the use cases definition process.

From this experience, we now understand better how to carry out the use cases also for the ASTRI mini-array, proposed for the CTA Observatory.

ACKNOWLEDGMENTS

This work is supported by the Italian Ministry of Education, University, and Research (MIUR) with funds specifically assigned to the Italian National Institute of Astrophysics (INAF) for the Cherenkov Telescope Array (CTA), and by the Italian Ministry of Economic Development (MISE) within the "Astronomia Industriale" program. We acknowledge support from the Brazilian Funding Agency FAPESP (Grant 2013/10559-5) and from the South African Department of Science and Technology through Funding Agreement 0227/2014 for the South African Gamma-Ray Astronomy Programme. We gratefully acknowledge support from the agencies and organizations listed under Funding Agencies at this website: http://www.cta-observatory.org/.

This paper has gone through internal review by the CTA Consortium.

REFERENCES

[1] Pareschi, G, et al, for the CTA Consortium, "The dual-mirror Small Size Telescope for the Cherenkov Telescope Array", Proc.33rd ICRC, arXiv:1307.4962 (2013).

[2] Acharya, B. S., et al., The CTA Consortium, "Introducing the CTA concept", Astroparticle Physics 43, 3-18 (2013).

[3] Montaruli, T., Pareschi, G., Greenshaw T, "The small size telescope projects for the Cherenkov Telescope Array", Proc. 34th ICRC, arXiv:1508.06472 (2015).

[4] Catalano, O., Maccarone, M.C., et al, for the ASTRI Collaboration and the CTA Consortium, "The camera of the ASTRI SST-2M prototype for the Cherenkov Telescope Array", Proc. SPIE 9147, doi: 10.1117/12.2055099, (2014).

[5] Vercellone, S., for the ASTRI Collaboration and the CTA Consortium, "The ASTRI mini-array within the Cherenkov Telescope Array", Proc. RICAP 2014, EPJ-EDP Sciences, arXiv: 1508.00799, (2015).

[6] Tosti, G., Schwarz, et al, , for the ASTRI Collaboration and the CTA Consortium, "The ASTRI MASS Software System", Proc. SPIE 9152, doi: 10.1117/12.2055067, (2014).

[7] Kulak, D., Guiney, E., "Use Cases: Requirements in context", Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©2003 ISBN-13: 978-0321154989

[8] Fowler, M., "UML distilled: A Brief Guide to the Standard Object Modeling Language", , Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©2003 ISBN:0321193687