Implementation of agile methodologies and model-based systems engineering for the management, design and development of a low earth orbit cubesat mission.

Juan Jose Mejia Gonzalez

Degree work to obtain the title of Aerospace Engineer

Internal Advisor
Juan Francisco, Puerta Ibarra, MSc.

External Advisor
Jeimmy Nataly Buitrago Leiva, MSc.

Universidad de Antioquia
Faculty of Engineering
Aerospace Engineering
Carmen de Viboral, Antioquia, Colombia
2024

Cita	Mejia Gonzalez, 2024 [1]
Reference IEEE (2020)	[1] Mejia Gonzalez, J. J. “Implementation of agile methodologies and model-based systems engineering for the management, design and development of a low earth orbit cubesat mission.”, Bachelor’s degree project, Aerospace Engineering, Universidad de Antioquia, Carmen de Viboral,Antioquia,Colombia,2024.

Dedicatoria

A mi madre Rosalba y a mi padre William, que siempre han estado al lado mio apoyándome. Gracias por darme todo de sí mismos, no solo para que pudiera alcanzar mis metas académicas, sino también para que me convirtiera en una persona de bien y para lograr todos mis sueños. Gracias por caminar a mi lado en cada paso de este viaje, siempre ofreciéndome una palabra de aliento en los momentos difíciles y celebrando cada uno de mis logros como si fueran los suyos. Es un honor y un privilegio llamarlos mis padres.

Agradecimientos

Quiero agradecerle a la tia Edith, ese apoyo incondicional y sin esperar nada a cambio ha sido fuente de inspiración y de aprendizaje, y me ha facilitado este logro académico.

A todos los profesores que han sido parte de mi formación académica. Gracias por compartir su invaluable conocimiento y sabiduría, en especial a Elias Montoya y Diego Hidalgo, cuya dedicación y entusiasmo han enriquecido profundamente mi experiencia educativa.

Un agradecimiento especial a mi asesor, Juan Francisco, por su guía experta, por creer en mis capacidades y por brindarme la oportunidad de desarrollar este trabajo. Su confianza y apoyo han sido cruciales para mi desarrollo profesional y personal. También a Jeimmy Nataly por su papel crucial en este trabajo, por su guia y enseñanzas, porque tu ética de trabajo, tu entrega y pasión son un faro de inspiración para mi, te admiro muchísimo.

No puedo dejar de mencionar a mis queridos amigos de la universidad: Santi, Samu, Sebas, Mario, Angelica, Laura, Nico, Camilo, Jorge y a todos mis compañeros de los equipos Delta V, Voyager y ASTRA. Gracias por acompañarme en este camino, por las risas, por esas horas de estudio, por los trabajos hechos en conjunto que nos han hecho crecer y soñar y por todos los momentos que nos han unido.

Finalmente, quiero agradecer a la Universidad de Antioquia por ser el escenario de esta maravillosa etapa de mi vida, proporcionándome las herramientas y oportunidades para crecer y prosperar.

Cada uno de ustedes ha dejado una huella imborrable en mi vida, y por eso, mi gratitud es infinita.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Acronyms and Abbreviations

ADCS	Attitude Determination And Control System
ConOps	Concepts of operations
DECO	Decommissioning
ECSS	European Cooperation for Space Standardization
EPS	Electrical Power System
FAC	Colombian Aerospace Force
FOP	Full Operations
GNC	Guidance, Navigation Control
GS	Ground Station
IOP	Initial Operations
ISS	International Space Station
IVVQ	Integration, Validation, Verification, and Qualification
JAXA	Japan Aerospace Exploration Agency
LA	Logical Architecture
LEOP	Launch and Early Orbit Phase
MBSE	Model-Based Systems Engineering
OBC	On-Board Computer
RAAN	Right Ascension of the Ascending Node
SE	Systems Engineering
SpOC	Space Operations Center
TT&C	Telemetry, tracking, and control
UNOOSA	United Nations Office for Outer Space Affairs

ABSTRACT

This work presents a case study on the integration of Agile Systems Engineering methodologies in the preliminary design phase of satellite systems, focusing on the ^{3}ColStar satellite mission. Through Model-Based Systems Engineering (MBSE), technical consistency was rigorously managed across various architectural models, ensuring coherency, and minimizing errors. Furthermore, with the implementation of the Arcadia Method, supported by the Capella modelling tool, the preliminary design was developed, and the use of digital engineering tools such as GMAT, Maltab/Simulink and python for validation and verification allowed the digitalization of the system represented in models that contain requirements, architecture, and the interfaces between the parts of the system. At the same time, the preliminary design process was streamlined and completed within an accelerated timeframe of 4 months, with weekly sprints driving progress based on the scrum methodology. This case study highlights the effectiveness of Agile Systems Engineering principles in enhancing the accuracy, communication, and efficiency of satellite systems preliminary design, providing valuable insights for future missions.Moreover, an adapted scrum framework is designed and proposed for the management of the following phases of the project.

***Keywords* — Model Based Systems engineering, Agile, Satellite, Digital engineering**

RESUMEN

Este trabajo presenta un estudio de caso sobre la integración de metodologías de ingeniería de sistemas ágiles en la fase de diseño preliminar de sistemas de satélite, centrándose en la misión del satélite ^{3}ColStar. A través de la Ingeniería de Sistemas Basada en Modelos (MBSE), se gestionó rigurosamente la consistencia técnica a través de varios modelos arquitectónicos, asegurando la coherencia y minimizando los errores. Además, con la aplicación del método Arcadia, apoyado en la herramienta de modelado Capella, se desarrolló el diseño preliminar, y el uso de herramientas de ingeniería digital como GMAT, Maltab/Simulink y python para la validación y verificación permitió la digitalización del sistema representado en modelos que contienen los requisitos, la arquitectura y las interfaces entre las partes del sistema. Al mismo tiempo, el proceso de diseño preliminar se agilizó y completó en un plazo acelerado de 4 meses, con sprints semanales que impulsaban el progreso basados en la metodología scrum. Este estudio de caso pone de relieve la eficacia de los principios de la ingeniería de sistemas ágil para mejorar la precisión, la comunicación y la eficiencia del diseño preliminar de sistemas de satélites, lo que proporciona información valiosa para futuras misiones.Además, se diseña y propone un framework scrum adaptado para la gestión de las siguientes fases del proyecto.

Palabras clave **— Ingenieria de Sistemas Basada en Modelos, Agile, Satelite, Ingenieria Digital**

I. INTRODUCTION

In recent decades, the field of small satellite engineering has undergone a transformative evolution, reshaping space science, communication, earth observation, and education. This revolution has been facilitated by the widespread availability and miniaturization of low-cost electronics, coupled with increased access to launch opportunities [9]. What was once solely the domain of governments and large organizations has now become democratized, with small companies, universities, and even low- and middle-income countries actively participating in satellite development [10]. Despite this progress, many satellite missions continue to face challenges, including delays, budget overruns, and suboptimal performance. CubeSat projects in which universities are involved and students make part of teams, in particular, struggle with issues such as high turnover rates, knowledge management, and balancing academic coursework with project responsibilities [11] [12].

In response to these challenges, there has been a growing interest in the adoption of agile methodologies and Model-Based Systems Engineering (MBSE) techniques within the field of engineering [13]. Agile methodologies, originally developed for software development [14], emphasize iterative and adaptive approaches, enabling teams to respond rapidly to changing requirements and feedback from stakeholders. This shift toward agility in engineering processes holds promise for streamlining workflows, optimizing resource allocation, and improving overall project outcomes, particularly within the dynamic context of small satellite development [9].

A careful review of the current state of the research field reveals a growing body of literature exploring the application of agile methodologies and MBSE in various engineering domains, including aerospace and satellite systems. Key publications such as [15] [16] [17] [18] have shown the benefits of adopting these approaches, highlighting their effectiveness in managing complexity, mitigating risks, and improving project outcomes.

In this work, the author explore how an integrated approach to agile Systems Engineering and Project Management can address the unique challenges faced by small satellite engineering teams. Through a detailed examination of the "^{3}ColStar"satellite mission, developed collaboratively by the Colombian Aerospace Force,Colombian universities and other international Institutions, the study demonstrates the application of agile methodologies and MBSE in the optimization of the development process [15]. Using these methodologies, the ^{3}ColStar mission team aim to accelerate time-to-market, reduce costs, and foster innovation in the small satellite industry, paving the way for future advancements in CubeSat technology.

This work is organized as follows: Section I provides a description of the ^{3}ColStar KiboCUBE Colombia mission and includes the Concept of Operations. Section III contains the concepts of Agile methodologies, Digital Engineering Model Based System Engineering, CubeSat missions, Arcadia Method and Capella software.Section IV comprises the constraints, Mission Requirements, Concept of Operations, Mission Architecture using MBSE, Validation and Verification using digital engineering tools, and Risk Analysis. Section V explains the applied agile methodology along with the overall management of the ^{3}ColStar KiboCUBE mission. Section VII discusses the proposed ^{3}ColStar KiboCUBE Systems Engineering Structure and outlines potential future research directions stemming from this work. Finally, Section VI summarizes the conclusions.

A. ^{3}ColStar satellite mission

The ^{3}ColStar KiboCUBE Colombia CubeSat (1U) mission (Fig. 1 and 2) emerged from the ambitious initiative to design a satellite manufactured up to 70 % in Colombia. This pioneering endeavor was made possible through the KiboCUBE call, organized jointly by the Japan Aerospace Exploration Agency (JAXA) and the United Nations Office for Outer Space Affairs (UNOOSA) [19]. This initiative provided an invaluable opportunity to develop a CubeSat and deploy it from the International Space Station (ISS) Japanese module ”Kibo”, thereby contributing to the sustainability and advancement of future space activities.

Based on the KiboCUBE opportunity, the constraints for the mission are as follows [19]:

- The satellite must be fully tested for the launch procedure according to the JAXA JEM Payload Accommodation Handbook.

- Deployment from the Kibo module in the ISS determines the mission orbit in semi-major axis, eccentricity, inclination, Right Ascension of the Ascending Node (RAAN), and argument of perigee. Mission duration needs to be analyzed and simulated to determine several important values, such as the amount of data produced from the payloads, solar radiation values, attitude determination and control, and communication link budgets, among others.
- The size of the CubeSat must not exceed 1U standards.
- Hazardous materials must not be used on the satellite since it will be deployed from a space crewed research facility such as the ISS.
- The expected profile of applicants is composed of government organizations, research institutes, universities, and other public organizations.

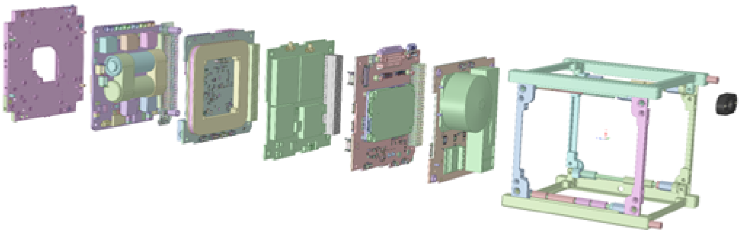


Fig. 1. Visualization of each of the satellite components (horizontal view) [1].

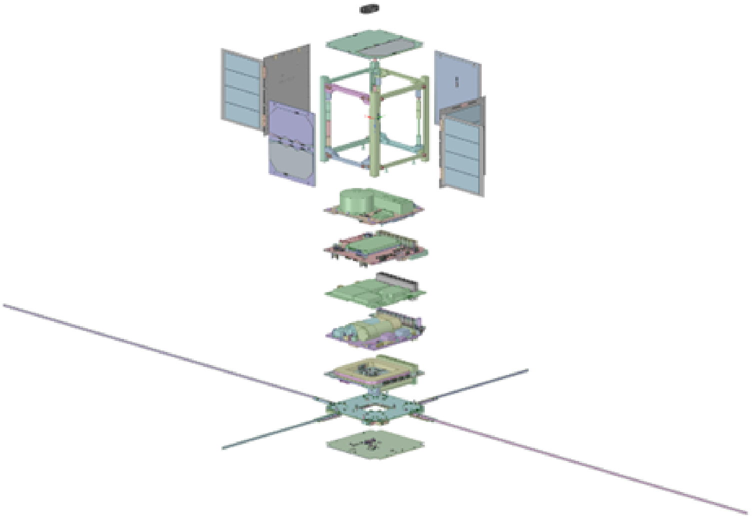


Fig. 2. Visualization of each of the satellite components (Vertical view) [1].

This mission stands at the intersection of academia, industry, and government, fostering collaboration among stakeholders in space exploration to drive scientific and social impact.

Equipped with two payloads, the CubeSat will contribute to scientific and technological advancements. Its primary payload features the MiniPIX TPX3 SPACE [20] sensor device, featuring a compact radiation camera tailored specifically for space missions. Designed to fit the CubeSat 1U platform, this device enables advanced particle tracking with minimal power consumption and weight. Its capabilities are crucial for monitoring particles generated by solar storms and mitigating potential damage to critical infrastructure, such as power lines, internet networks, and satellites [20]. Provided by ADVACAM, the device ensures precise particle characterization and real-time analysis, thus enhancing the scientific objectives of the mission [1].

Additionally, the CubeSat will incorporate a secondary payload comprising an IoT device designed for data transmission to a mobile ground station. This facilitates analysis and risk control within the Internet of Things application system. Furthermore, the satellite features an in-house developed Fine Sun Sensor and a proof-of-concept for the research and development of a reaction wheel and magnetorquers [1].

Beyond its scientific goals, the mission plays a pivotal role in advancing Colombian expertise in space components and subsystems, including structure, EPS, OBC, ADCS, and electronics. Collaboration involving 13 institutions (Figure 11), encompassing professors, researchers, and students at various academic levels, fosters national and international cooperation, thereby enhancing Colombia's capabilities in space technology [1] (Figure 8).

II. OBJECTIVES

The objectives outlined by the author for his participation in this project are as follows:

A. General Objective

To implement agile methodologies and Model-Based Systems Engineering (MBSE) for the management, design and development of a low earth orbit CubeSat mission, with the purpose of efficiently optimizing the processes associated with the specific space development of Cubesat-type missions.

B. Specific Objectives

- Define agile methodologies and Model Based Systems Engineering (MBSE) tools for the satellite mission, in order to optimize the management and establish the requirements and constraints of the project.
- Develop a model that comprehensively represents the system architecture and subsystems of Cubesat, allowing a clear and accurate visualization of its structure.
- Use simulation tools, such as GMAT, STK, Python or MATLAB, to verify and validate the requirements and needs of the CubeSat system and subsystems.
- Perform a comprehensive risk analysis to identify potential challenges during CubeSat development and propose effective mitigation strategies.
- Validate satellite subsystems such as the CubeSat power subsystem and attitude control subsystem through MATLAB/Simulink simulations, addressing modes of operation such as detumbling and ensuring system efficiency under various conditions.
- Obtain as final results a detailed model of the CubeSat, the complete architecture of the mission and the satellite, as well as the ability to effectively validate and verify the requirements and needs initially defined.

III. LITERATURE REVIEW

A. Agile Methodologies

Agile methodologies are a collection of iterative and incremental software development approaches that emphasize flexibility, adaptability to change, frequent delivery of working software, and close collaboration between development teams and stakeholders. Key principles include prioritizing customer satisfaction through early and continuous delivery of valuable software, welcoming changing requirements (even late in the development cycle), delivering working software within short intervals, fostering close collaboration between business stakeholders and developers, and building projects around motivated individuals who are given the support and trust to deliver. Face-to-face conversation serves as the primary communication method, with working software as the key measure of progress. Agile methods also prioritize sustainable development, continuous attention to technical excellence, simplicity, self-organizing teams, and regular reflection for process improvement.

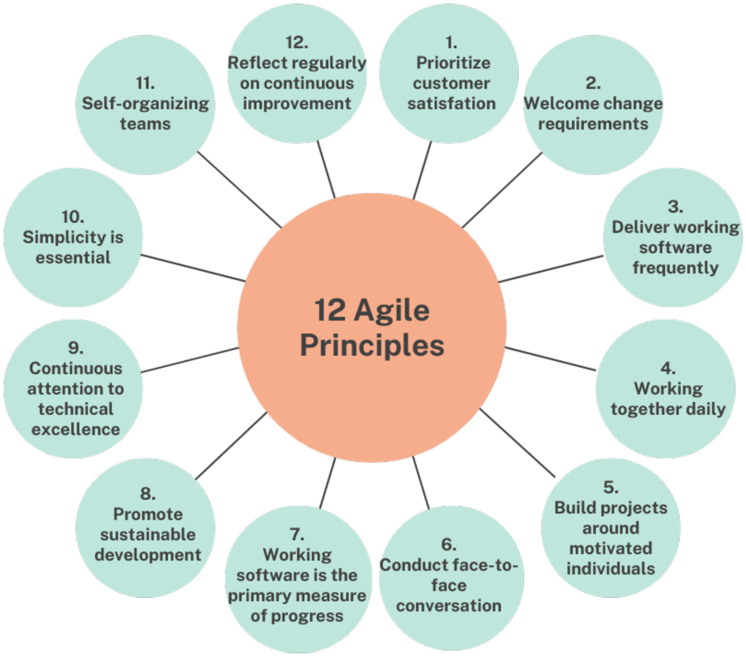


Fig. 3. the 12 agile principles in Agile Manifesto [2].

B. Model Based Systems Engineering

MBSE is an approach that uses digital models of the system and its engineering aspects as the main way to share and manage information, feedback, and requirements, instead of relying on documents. It covers the whole process of creating, communicating, and ensuring that all the digital models that describe a system are consistent from the conceptual design phase through the later phases of the life cycle, such as to requirements definition, design, analysis, and verification and validation activities [21]. MBSE is based on modeling languages and methods such as Systems Modeling Language (SysML) [22], which is used in tools like Cameo Systems modeler [23], MagicDraw [24] [25] or the Arcadia method [26], used in Capella. These MBSE tools allow representing and communicating the structural, functional and dynamic aspects of a complex system and it aims to improve the efficiency, quality and traceability of the systems engineering process, as well as to facilitate collaboration between the different actors involved [27].

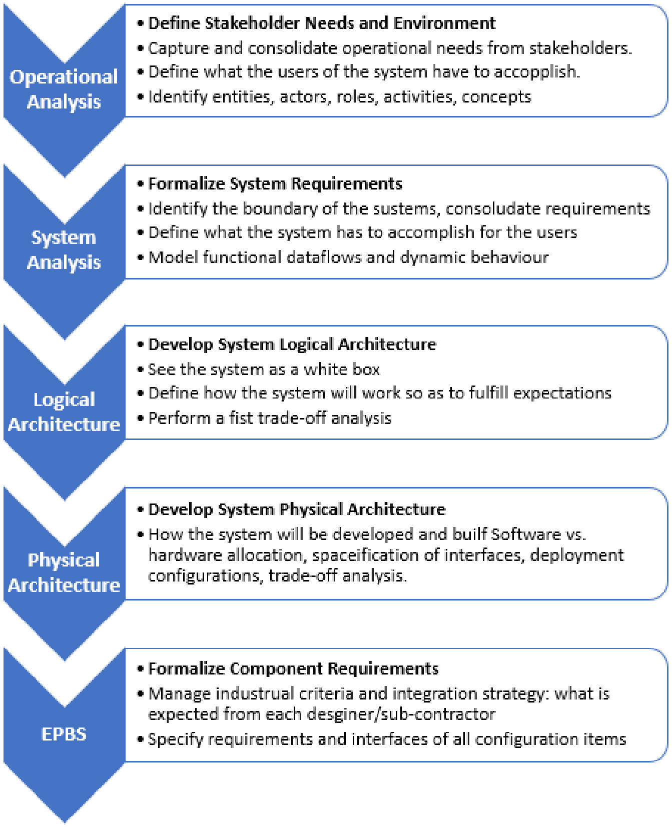
C. Arcadia method and Capella modelling tool

Arcadia enables thorough modeling of complex systems in the Architecture Engineering context, across multiple levels of abstraction. It is founded on a hierarchical framework that first defines the problem space at the top level, and later defines proposed solutions that traverse the system's various elements. It is bolstered by a viewpoint-centric approach that underscores the need to integrate the many views that are vital to the design of a system. It is further reinforced by its support for a thorough trade-off analysis that allows decisions at all levels of architectural design [26] [28] [29].

Arcadia is a tooled method devoted to systems & architecture engineering, supported by the Capella modeling tool. This Method is presented in Figure 4, and it describes the detailed reasoning to:

- Understand the real customer needs.
- Define and share the product architecture among all engineering stakeholders.
- Early validate its design and justify it.

- Ease and master Integration, Validation, Verification, and Qualification (IVVQ).



- **Define Stakeholder Needs and Environment**
- Capture and consolidate operational needs from stakeholders.
- Define what the users of the system have to accopplish.
- Identify entities, actors, roles, activities, concepts

- **Formalize System Requirements**
- Identify the boundary of the sustems, consoludate requirements
- Define what the system has to accomplish for the users
- Model functional dataflows and dynamic behaviour

- **Develop System Logical Architecture**
- See the system as a white box
- Define how the system will work so as to fulfill expectations
- Perform a fist trade-off analysis

- **Develop System Physical Architecture**
- How the system will be developed and builf Software vs. hardware allocation, spaceification of interfaces, deployment configurations, trade-off analysis.

- **Formalize Component Requirements**
- Manage industrual criteria and integration strategy: what is expected from each desginer/sub-contractor
- Specify requirements and interfaces of all configuration items

Fig. 4. Arcadia method for System Architecture Development.

It can be applied to complex systems, equipment, software, or hardware architecture definition, especially those dealing with strong constraints to be reconciled (cost, performance, safety, security, reuse, consumption, weight. . .).

It is intended to be used by most stakeholders in the definition of the system / product / software or hardware, and IVVQ is intended to be used as a common reference and collaboration support.

The Capella modeling tool is used to define the entities involved in the project, its hierarchy, and capabilities from the Operational Level, and through several systems engineering decision-making criteria, realizing the different analysis until the physical description of the satellite and how it connects to the space and ground segments. The mission's core analyses focus on the Logical and Physical Architecture, with the Operational and System architecture laying the foundational framework. In this setup, the system's logical functions sets the stage for the subsequent physical specifications. This approach ensures that while technology may evolve and become outdated, the defined functionalities remain central, guiding the behavior and integration of each system component effectively.

D. Digital Engineering

Digital engineering is an integrated approach to the design, development, and lifecycle management of systems that heavily utilizes digital models, simulations, data analytics, artificial intelligence, and collaborative technologies. It aims to replace traditional document-heavy engineering practices with more streamlined and data-driven workflows. The goals of digital engineering include increased efficiency and innovation through faster design iterations, improved decision-making based on insights from digital models and real-world data, enhanced collaboration and knowledge sharing across disciplines, better risk management by exploring scenarios in a virtual environment, and overall system optimization using data analytics and AI algorithms.[30] [31] and [32] are examples of the use of digital engineering in the industry.

E. CubeSat

CubeSats are a class of miniaturized satellites, typically based on a standardized form factor of 10cm x 10cm x 10cm units (1U) [33] . They have revolutionized space access due to their affordability, faster development times, educational opportunities, and potential for technology demonstration. CubeSats provide lower launch and manufacturing costs, enable quicker iterations due to simplified design, allow universities and even high schools to launch their own satellites, and serve as platforms to test new technologies and concepts in a relatively low-risk space environment [34].

Aditionally, Satellite missions involve the design, development, launch, and operation of artificial satellites placed into orbit around Earth or other celestial bodies. These missions serve a wide range of purposes, including Earth observation (collecting imagery and data about Earth's atmosphere, land, oceans, and weather patterns), communications (providing telecommunications services like television broadcasting, internet access, and phone calls), navigation (operating GPS and other satellite navigation systems), scientific research (conducting experiments in the unique environment of space), and military operations (supporting national defense with reconnaissance, surveillance, and secure communications) [35] [36] some are examples of these missions .

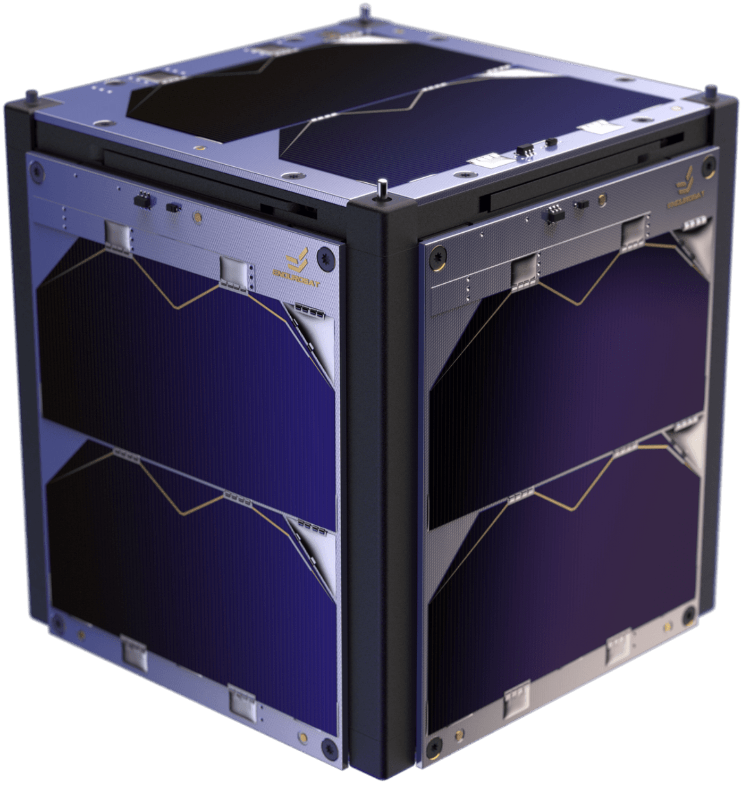


Fig. 5. Endurosat 1U CubeSat [3].

CubeSats, despite their small size, contain essential subsystems to function in space. These include:

- Electrical Power System (EPS): Responsible for generating, storing, and regulating power for all satellite components. This often involves solar panels, batteries, and power distribution units.

- Attitude Determination and Control System (ADCS): Orients the CubeSat correctly and maintains its stability in space. ADCS often uses sensors (like magnetometers and sun sensors), actuators (like reaction wheels or magnetorquers), and control algorithms.
- Telemetry, Tracking, and Command (TTC): Facilitates communication with ground stations for sending telemetry data (health and status of the satellite), receiving commands, and tracking the CubeSat's position.
- On-Board Computer (OBC): The "brain" of the CubeSat, the OBC processes sensor data, runs control algorithms, manages other subsystems, and executes commands from the ground. Each of these subsystems plays a crucial role in the overall success of the CubeSat mission.

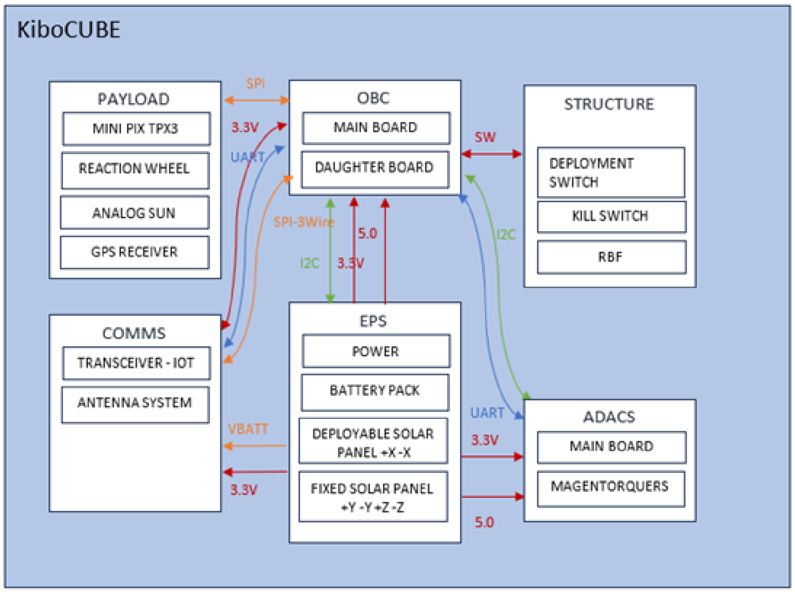


Fig. 6. [3]ColStar General structure of the subsystems.

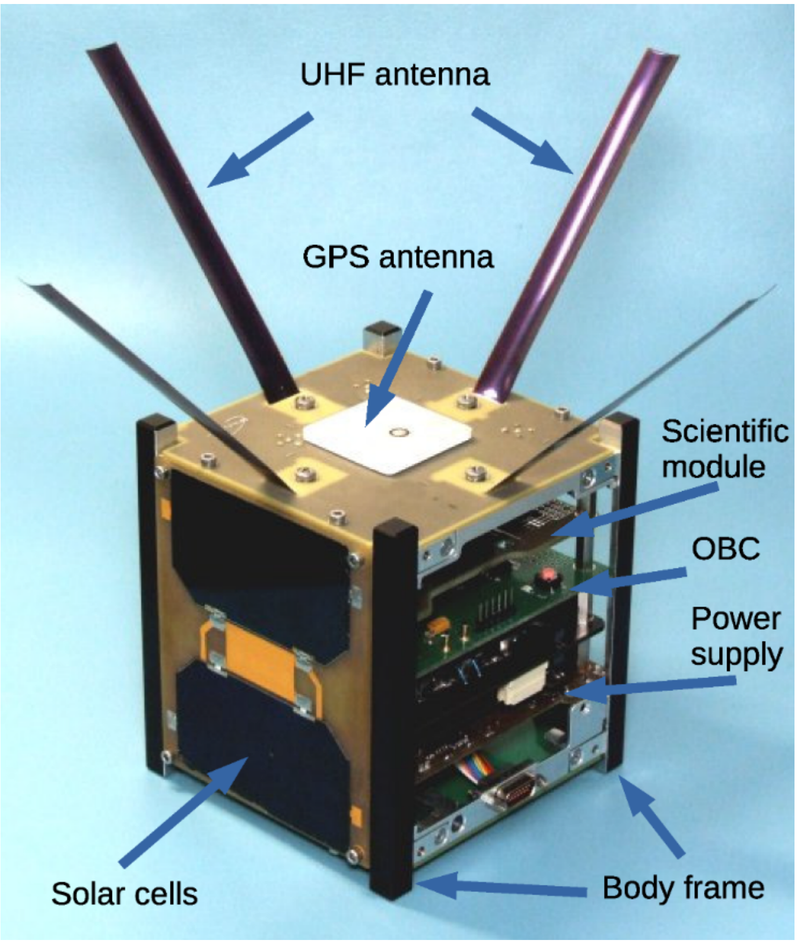


Fig. 7. CubeSat subsystems [4]

.

IV. COLSTARCUBE MISSION

A. Constraints

To design a 1-unit (1U) CubeSat that complies with the standards and requirements necessary for its deployment through the JEM Small Satellite Orbital Deployer (J-SSOD) on the International Space Station (ISS), several technical specifications and regulatory guidelines must be considered. The design restrictions and requirements for the CubeSat are derived from [reference]. A summary of these is presented below:

1. General Dimensions: Size (Width x Length x Height): 100 mm x 100 mm x 113.5 mm (+/-0.1 mm). Document: JX-ESPC-101133-E.

2. Mechanical Interfaces (Document: JX-ESPC-101133-E.)
    - Rails: Must have a minimum width of 8.5 mm and cannot have a roughness greater than Ra 1.6 m. Additionally, the rails must be treated with hard anodizing after machining.
    - Separation Force: The CubeSat must be capable of withstanding separation forces without suffering damage or misalignment.

3. Electrical Interface (Document: JX-ESPC-101133-E.): Deployment Switch: Necessary to ensure that the CubeSat remains inactive during launch and is activated only after deployment.

4. Environmental Requirements(Document: JX-ESPC-101133-E)
    - Vibration and Shock: The CubeSat must meet specified vibration and shock levels for launch and deployment.
    - Temperature and Vacuum: Must be designed to operate within the range of temperatures and in the vacuum of space.

5. Safety and Compatibility Requirements (Documents: JSC-20793, SSP51721, SSP52005 for general and specific hardware safety.):

- Material and Process Control: Must comply with standards to prevent contamination and ensure compatibility with the ISS environment.
- Safety Analysis: Including risk assessment and mitigation.

6. Outgassing(Document: ASTM-E595-84):

   - Gas Emission: The CubeSat must meet gas emission requirements to prevent contamination in space.

7. Verification and Validation Process(Document: JX-ESPC-101133-E.): There must be a verification and validation process to ensure all requirements are met before launch.

8. Documentation and Approvals(Documents: JX-ESPC-101133-E and applicable from the launching agency JAXA):

   - Complete Technical Documentation: Including designs, analyses, test results.
   - Necessary Approvals: From the initial proposal to the final approval for launch.

B. Mission Objectives

The goal of is mission is to harness the potential of a 1U CubeSat for a dual purpose: enhancing solar wind particle measurement for a deeper understanding of space weather impacts and pioneering the integration of NB-IoT technology for IoT applications in remote areas. This endeavor aims to bolster Colombia's expertise in space technology and IoT infrastructure, fostering innovation and capacity-building in underserved communities.

Using the SMART criteria (Specific, Measurable, Achievable, Realistic, Time-bounded) to define what the team wants to achieve through the project, CubeSat Development and Deployment Contribute to Capacity-Building with the following:

1. Specific:

   - Solar Observation: Measure solar wind particles to improve understanding of space weather.

- NB-IoT Technology for IoT: Test and demonstrate NB-IoT in space for IoT applications, particularly in remote monitoring of soil conditions.

2. Measurable:
   - Track the accuracy and consistency of solar wind data collected
   - Evaluate the performance and reliability of NB-IoT technology for space-based IoT applications.
3. Achievable: Utilize established expertise in solar observation and collaborate with IoT technology specialists to implement and test NB-IoT technology.
4. Realistic: The project leverages current technological advancements and partnerships, making these goals attainable within the scope of national capabilities.
5. Time-Bounded: Completion of the CubeSat development, launch, and operational phases within a defined timeline, ensuring timely data collection and analysis.

C. Requirements

1) Mission Requirements

TABLE I

PRIMARY AND SECONDARY MISSION REQUIREMENTS

Req ID	Requirement	Rationale
PrimMis-001	The satellite must obtain solar weather observation data and facilitate the transmission of data from IoT hubs on the ground.	The satellite in each nominal mode will have 30 minutes of activation in which it will collect 197 MB corresponding to the sensor images for space radiation, the sensor will receive a total of 1800 images during this mode

PrimMis-002	The information vector must be transmitted using the satellite's communication frequency (amateur radio).	The data will be sent to the Ground stations using the assigned frequency in such a way that it can be heard by the selected ground stations
PrimMis-003	The project should be developed mainly in Colombia and by students, professors and researchers from the country.	The team has multidisciplinary personnel with the capacity to support the different areas of the project
PrimMis-004	Develop and deploy a CubeSat capable of performing solar observation, focusing on the measurement of solar wind particles and IoT sensors.	Joint work between the different institutions for the design and development of the electronics for the solar climate sensor and a ground deployment of sensors for the measurement of environmental variables.
PrimMis-005	Ensure compliance with Space Debris Mitigation Guidelines to minimize space debris generation and adhere to responsible space practices.	Strict adherence to Space Debris Mitigation Guidelines is vital to minimize debris generation, implementing measures such as controlled reentry and responsible design practices for sustainable space use.
SecMis-001	Create an online repository of project documentation, including CubeSat design schematics, communication protocols, and data analysis methods.	Generation of a specific mission center to place the technical material in an understandable didactic form f or different sectors of society to consult online.

SecMis-002	Publish research findings and best practices in CubeSat development, NB-IoT technology, and IoT applications.	It is necessary to create a scientific mission center to be able to share all the advances, tests and lessons learned that are being developed throughout the project, this will serve as support for the community but to a greater extent for the entire development team, with the accompaniment of international experts. (3ColStar mission center)
SecMis-003	Establish partnerships with local industry stakeholders to promote knowledge exchange and resource-sharing in space technology and IoT	There is support from different actors belonging to the sectors of academia, state and industry that are part of the triple helix that have developed important advances individually and it is necessary to achieve integration.
SecMis-004	Collaborate with local universities and research institutions to facilitate CubeSat development and capacity-building initiatives	Collaboration with local academic institutions enhances CubeSat development by combining expertise, resources, and fostering future talent.
SecMis-005	Organize capacity-building workshops and training sessions for local scientists, engineers, and students.	It is necessary to create a scientific community capable of providing lectures, workshops and training to people with different levels of education.

2) Satellite Design Requirements

TABLE II
DESIGN REQUIREMENTS

Req ID	**Requirement**	**Rationale**
Des-001	The satellite should be able to store the 196 MB of the solar weather sensor and the 90 MB generated by the IoT payload.	Ensures that the satellite's onboard data storage is sufficient to accommodate the data generated by the IoT sensors and the payload(. Adequate storage capacity ensures the successful collection and retention of crucial mission data.
Des-002	The project will develop a flight-enabled CubeSat capable of being launched from the International Space Station.	The CubeSat shall be compatible with being launched from the International Space Station (ISS). This requirement is crucial as it ensures the satellite is designed to meet the specifications and constraints associated with launch mechanisms and deployment systems aboard the ISS.
Des-003	The project must ensure that the satellite mission can be fully accomplished by making use of its subsystems.	The CubeSat's subsystems must collectively support the fulfillment of the satellite mission. Each subsystem (such as EPS, OBC, etc.) plays a vital role in ensuring the successful execution of the mission objectives. This requirement emphasizes the integration and functionality of these subsystems to accomplish the overall mission goals.

TABLE II
DESIGN REQUIREMENTS

Req ID	Requirement	Rationale
Des-004	The project shall comply with the KiboCube ICD (Interface Control Document) deployment standards.	Adhering to the KiboCube Interface Control Document (ICD) deployment standards ensures compatibility and smooth integration of the CubeSat with the deployment mechanisms and interfaces present on the ISS. Meeting these standards is crucial to ensure seamless deployment and operation in the ISS environment.
Des-005	The project shall comply with the safety standards of the International Space Station NSTS SSP 51700.	Compliance with safety standards set by the International Space Station (specifically NSTS SSP 51700) is paramount to ensure the CubeSat's design, materials, and operations do not pose any risks or hazards to the ISS, its crew, or other space assets. This requirement prioritizes the safety and reliability of the satellite's design and operation within the ISS environment.

TABLE II

DESIGN REQUIREMENTS

Req ID	**Requirement**	**Rationale**
Des-006	CubeSat must adhere to specified size and mass constraints for compatibility with ISS launch and deployment mechanisms.	Ensures safe handling and compatibility with ISS deployment systems.
Des-007	CubeSat needs reliable power sources, such as solar panels and energy storage (e.g., batteries), to ensure continuous operation.	Ensures sustained functionality throughout the mission.
Des-008	CubeSat requires reliable communication systems to transmit data effectively to and from Earth.	Facilitates data transmission for mission success.
Des-009	CubeSat must have thermal control mechanisms to regulate internal temperatures and protect components.	Preserves functionality and prevents damage due to extreme temperatures in space.
Des-010	CubeSat must meet specific orbital and stability criteria to fulfill mission objectives	Ensures proper operation and achievement of mission goals.

TABLE II
DESIGN REQUIREMENTS

Req ID	Requirement	Rationale
Des-011	CubeSat needs adequate control and handling systems for maneuvers, orientation adjustments, and stability.	Facilitates necessary adjustments and maneuvers during its time in orbit.
Des-012	CubeSat must be designed to withstand space conditions, including radiation, vacuum, temperature changes, and launch vibrations.	Ensures structural integrity and functional capability throughout the mission's duration.
Des-013	The sensors must provide high-quality and accurate data to facilitate credible scientific research, meeting the standards required for academic investigations.	Ensuring high-quality, accurate data is crucial for scientific credibility, enabling universities and students to conduct reliable and meaningful research.

3) Ground Segment Design Requirements

TABLE III

GROUND SEGMENT DESIGN REQUIREMENTS

Req ID	**Requirement**	**Rationale**
GSeg -001	The ground segment must ensure IoT connectivity and data collection in remote zones or areas not interconnected to IoT sensing systems.	Technology will be developed by the partner universities for narrowband transmission of IoT data for the Ground Satellite to Satellite uplink and Node IoT to Satellite. As a starting point for the implementation of satellite IoT technology in Colombia
GSeg -002	The ground station shall contain all the necessary devices to establish communication with the satellite.	Taking into account the losses present in the communication link, as well as the considerations in figure of noise and gains of the devices.
GSeg -003	Definition of the commands required for the control of the georeferenced telemetry provided by the satellite and for checking satellite subsystems.	For the security and correct operation of the satellite, commands will be defined to check the subsystems and status of the satellite, as well as the management of the information received by NB-IoT.

GSeg -004	The communication between satellite-ground stations complies with a frequency and bandwidth according to payload.	The ground station will have to operate in UHF band complying with the requirements, especially the transmission power regulated by the government to guarantee the correct access of the satellite. The communication shall have the ability to send and receive satellite data both IoT and Payload with the minimum error rate allowed and with a frequency within the amateur radio band.
GSeg -005	Geographical position of the ground station(s) present in a coordinated network and of the satellite	The ground station shall have specialized software for satellite tracking, as well as antenna rotation systems to establish communication link with the satellite.

4) Operational requirements

TABLE IV

OPERATIONAL REQUIREMENTS

Req ID	Requirement	Rationale
Ope-001	The satellite should be able to delete the stored data from the solar sensor and IoT module measurements and be ready for new storage.	Ensures continuous data collection by freeing up memory space, allowing the satellite to accommodate new measurements without interruptions due to storage limitations.

Ope-002	The CubeSat must be capable of controlled activation and deactivation to conserve power and operate efficiently as needed.	Enables energy conservation and controlled operational states for optimal functionality.
Ope-003	The Cubesat shall function in various operational modes (e.g., data collection mode, transmission mode, power-saving mode) as per mission requirements.	Provides versatility to adapt to different operational needs during the mission.
Ope-004	The Satellite shall be capable of controlled maneuvers to adjust orientation, or other orbital characteristics based on mission objectives.	Allows for necessary orbital adjustments for mission goals and objectives.
Ope-005	The Satellite shall be capable of synchronizing communication windows with Earth for data transmission and receiving commands during specific orbit periods.	Ensures effective communication and data exchange with ground stations at designated times.
Ope-006	The Satellite shall have ability to detect issues and respond to emergencies using backup systems or safety protocols.	Ensures robustness and ability to handle unexpected situations during the mission.

Ope-007	The Satellite shall be capable of self-diagnostics to detect potential failures and perform preventive maintenance.	Ensures continuous health monitoring and proactive maintenance to sustain operational capabilities.
Ope-008	The Satellite must follow a specific and controlled sequence for system and subsystem startup and shutdown to ensure operational integrity.	Prevents errors or damage during operational transitions and maintains operational stability.

D. Concept of Operations

A general operating framework has been defined, considering four stages:

1. STAGE 1: Launch and Early Orbit Phase (LEOP) operations: Comprise the satellite operations from launch through the early orbit phase. In this stage, the first contact with the ground station is established, and the satellite's Early Operation Test (Early Operation EOP) with the main and secondary payload are conducted.

2. STAGE 2: Initial Operations (IOP): Once it has been verified that everything is working properly and that the satellite is in optimal health conditions after stabilizing in orbit through the ADCS and ensuring a reliable power supply through the EPS, the stage of the initial operations of the payloads begins.

3. STAGE 3: Full Operations (FOP): At this stage, the satellite initiates its nominal operations, including payload health checks, communications, and data download runs.

4. STAGE 4: DECOMMISSIONING (DECO): Once the satellite has completed its operational mission, a decommissioning mode is initiated.

For each stage established, some modes of operation have been defined, which are mentioned below:

- Standby (Sb): Period before the satellite is turned on. All subsystems are inactive.
- Released (R): After the standby period, the satellite is turned on, with the EPS and OBC as the only active subsystems.
- Pre-detumbling (PD): Once the UHF antennas have been deployed, the EPS, OBC, COMMS and AOCS (only determination for telemetry) are active. The COMMS subsystem is transmitting data.
- Detumbling (D): Same as the pre-detumbling state, with the magnetorquers operating.
- Detumbled (Dd): Once the satellite is detumbled, the AOCS is keeping the desired attitude. The COMMS subsystem is transmitting data.
- Basic (Ba): The satellite transmits telemetry exclusively
- Nominal (N): Satellite is fully operative. The primary and secondary payloads are executed 1 times per orbit respectively. The COMMS subsystem is transmitting data.Inside the Nominal mode there are four key operations:
    1. Nominal IoT Dowlink (Nid): IoT application data download
    2. Nominal IoT Uplink (Nid): Receiving data from the IoT application
    3. Nominal Space Weather (Ns): Nominal operation of the Space Weather payload (ON).
    4. Nominal Space Weather Dowlink (Nsd): Space Weather payload data download.
- Decommissioning (DC): The satellite is deactivated and safely manage the end of its useful life. Typical activities in Decommissioning mode include:
    1. Systems Decommissioning: The satellite's operating systems, such as scientific instruments, transmitters, and other electronic components, are shut down in an orderly fashion
    2. Fuel and Battery Purge: Batteries are discharged to minimize the risk of explosions or malfunctions that could generate more debris.
    3. Final Transmission and Power Shutdown: A final transmission is sent and then the satellite power is permanently shut down.

The Concept of operations is illustrated in the Mission concept (Figure 8). Moreover, the modes of operation defined above are shown as a state machine diagram in Figure 9. The Nominal mode is shown in detail in Figure 10.

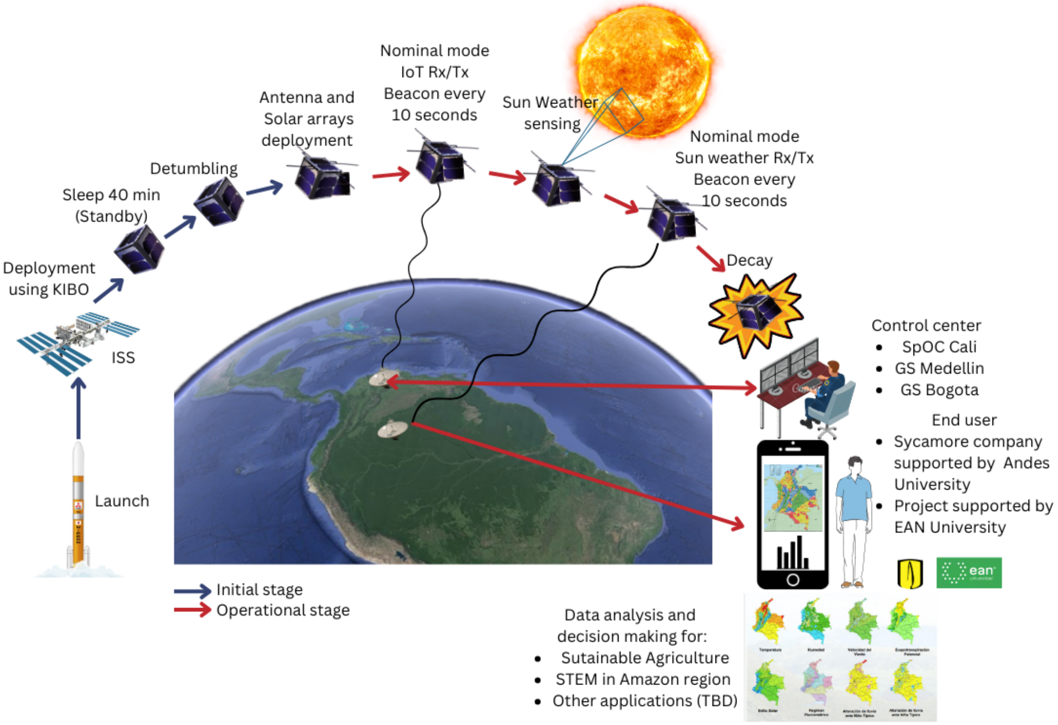


Fig. 8. KiboCUBE Team Colombia ^{3}ColStar Mission Concept.

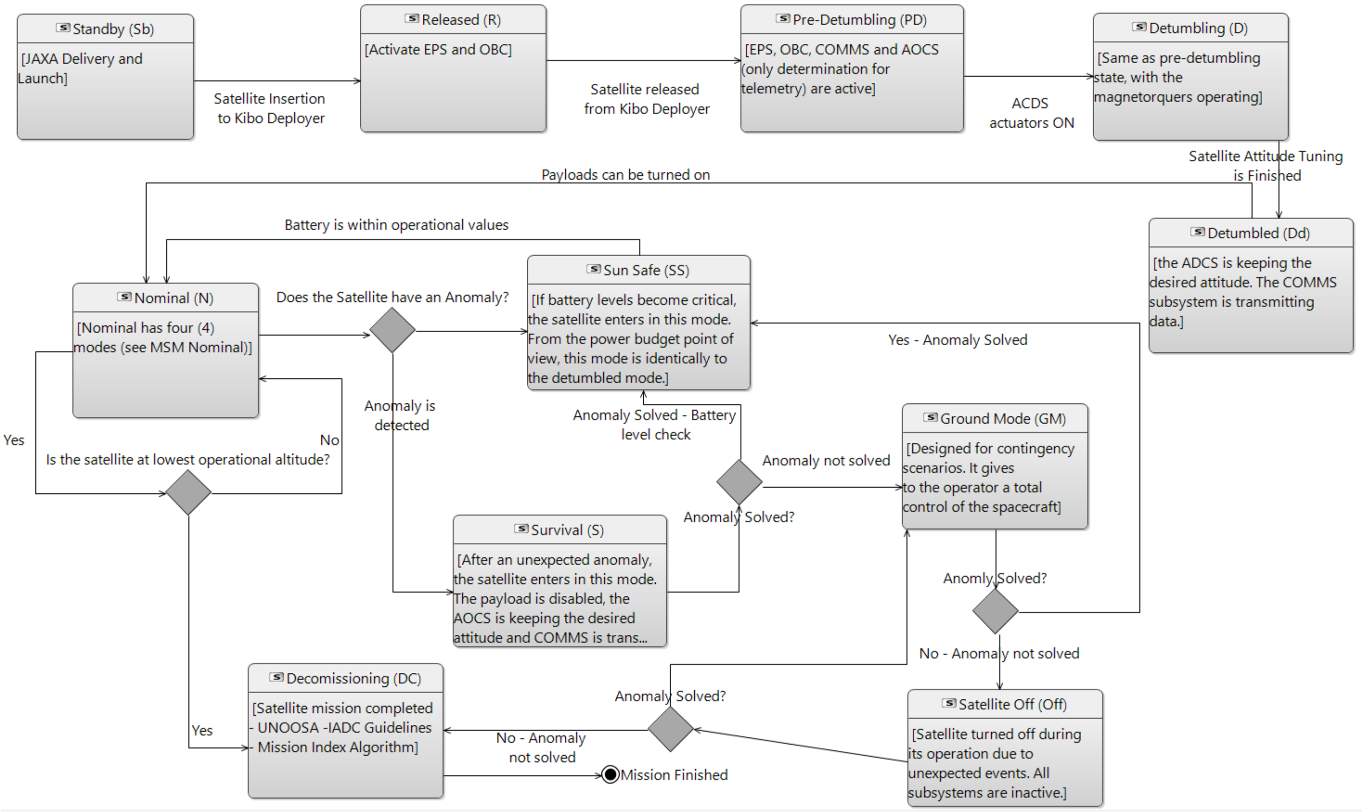


Fig. 9. Modes of operations of the ^{3}ColStar Mission Concept.

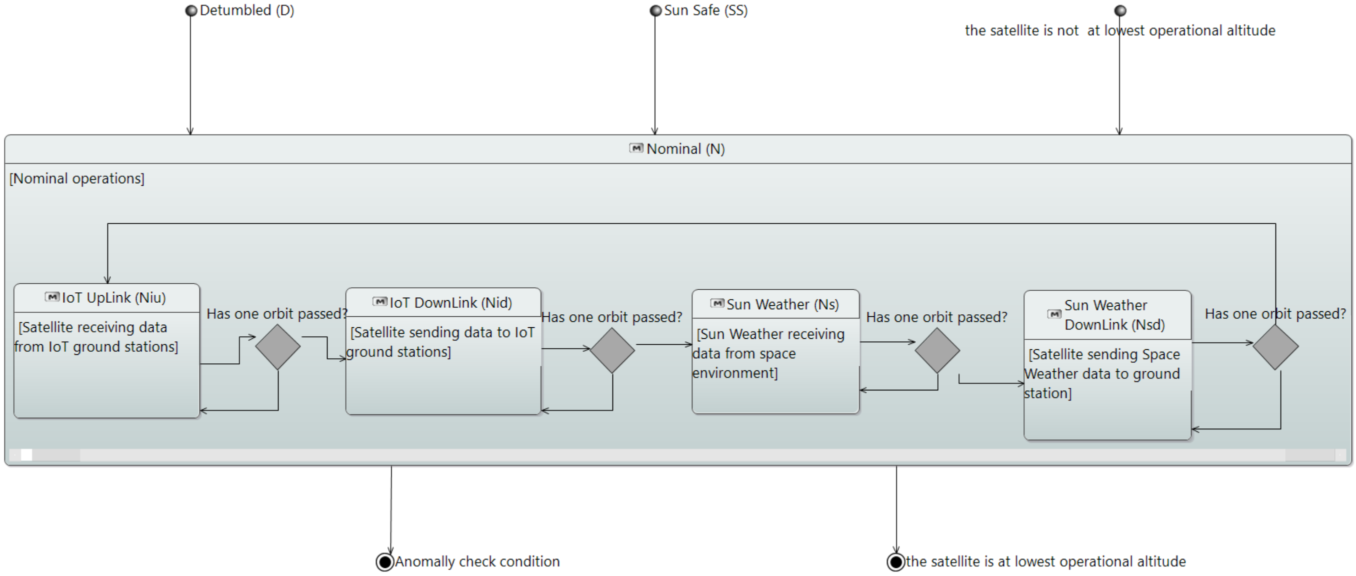


Fig. 10. Nominal mode.

E. Mission Architecture

There are two primary operational entities: the Aeronautics and Space Education Headquarters and the KiboCUBE project itself. Six operational entities are derived from the project such as Project management, Financial segment, Science Team, Systems Engineering, Subsystem Research and Development and Testing Facilities. Each of these entities has operational actors, which are represented with the silhouette of people in the figure. Within the Subsystem R & D entity, the actors that will be in charge of each subsystem of the satellite can be found, for example, the Universidad Distrital will be in charge of the OBC subsystem and the Universidad Nueva Granada of the thermal subsystem. Within the IoT & Education entity are the actors that design the satellite payload and the actors that will make use of it.

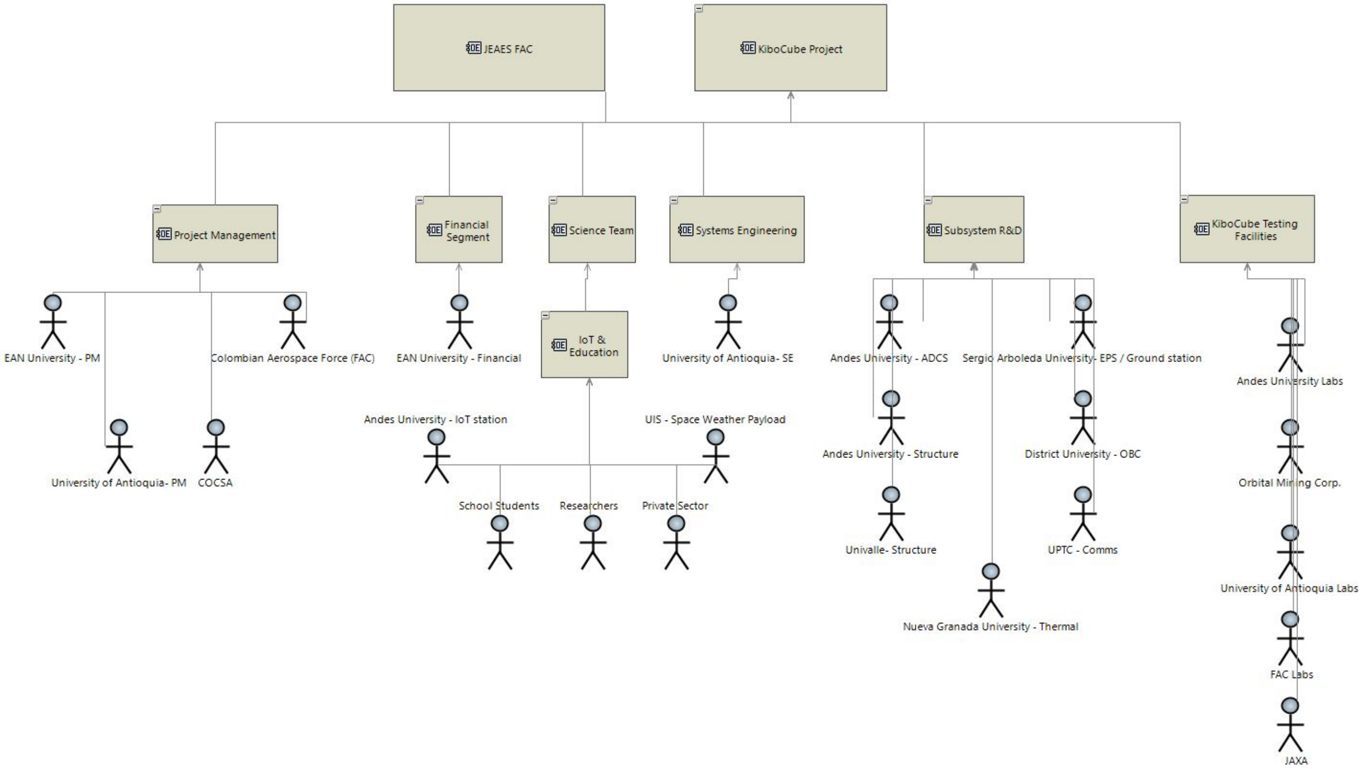


Fig. 11. Main Operational Entities of ^{3}ColStar KiboCUBE Project.

Figure 12 illustrates the main operational capabilities of the mission. From the mission itself, during the satellite lifetime and based on the two payloads presented (IoT and Space

Environment analyses), two capabilities are:

- Produce space environment data
- Produce IoT data

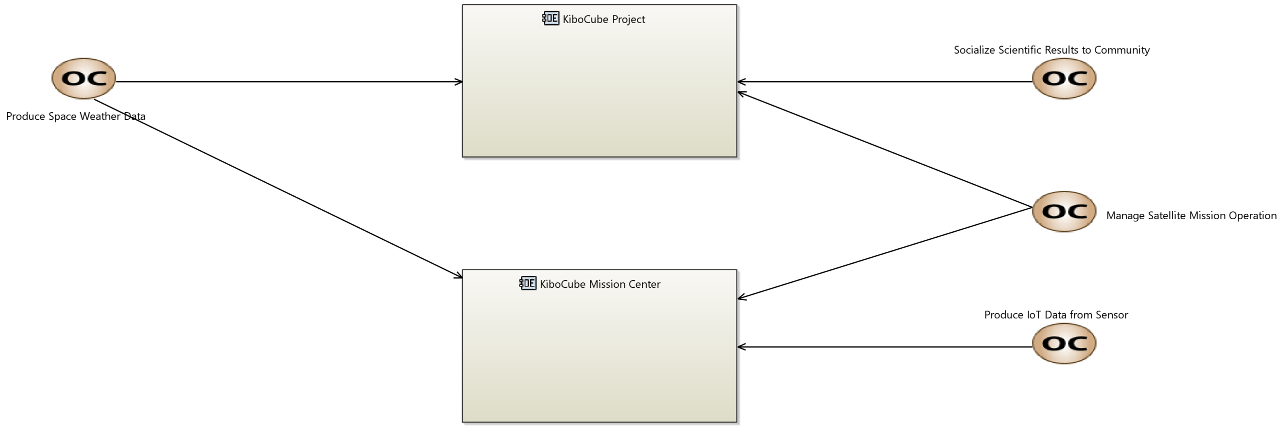


Fig. 12. Main Operational Capabilities of ^{3}ColStar KiboCUBE.

There are also other two capabilities, one is to operate the satellite and one of the most important is to share the results with the Colombian society and scientific community.

Figure 13 shows the logical architecture of the satellite,the principle of the Logical Architecture (LA) is to start to "open the box" by implementing the big decisions of the solution, in terms of principles of construction, and ways to fulfill the expectations of stakeholders; it is then formalized by means of a decomposition into abstract structural elements called Logical Components, this components force ourselves to exclude all technological consideration or implementation choice. In this diagram, the main objective is to show the principles of behavior and interaction between one logical component to another, in response to the previous needs.

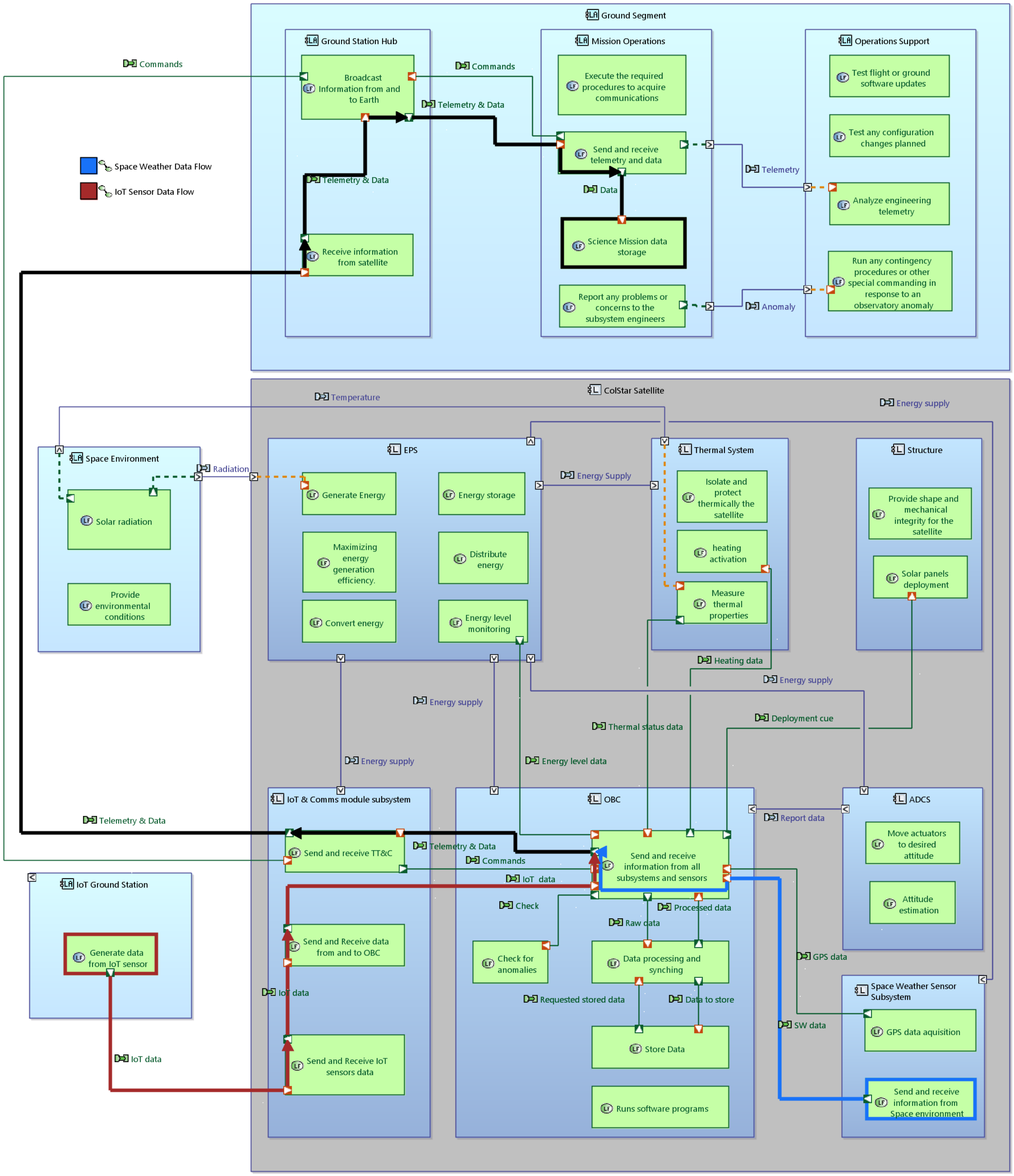


Fig. 13. [3]ColStar Logical Architecture Diagram.

Logical Component: Structural element within the System, with structural Ports to interact with the other Logical Components and the external Actors. A Logical Component can have one or more Logical Functions. It can also be subdivided into Logical subcomponents.

- Logical Actor: Any element that is external to the System (human or non-human) and that interacts with it.
- Logical Function: Behavior or service provided by a Logical Component or by a Logical Actor. A Logical Function has Function Ports that allow it to communicate with the other Logical Functions.
- Functional Exchange: A unidirectional exchange of information or matter between two Logical Functions, linking two Function Ports.

In this diagram this architecture is composed of the Ground segment, where there are 3 logical actors, the ground station hub, the mission operations and the operations support. Other logical actors such as the IoT ground station and the Space Environment are also shown. The logical component in which the diagram revolves around is the [3]ColStar Cubesat, within this are all the subsystems, their functions and how they interact with each other and with the logical actors.

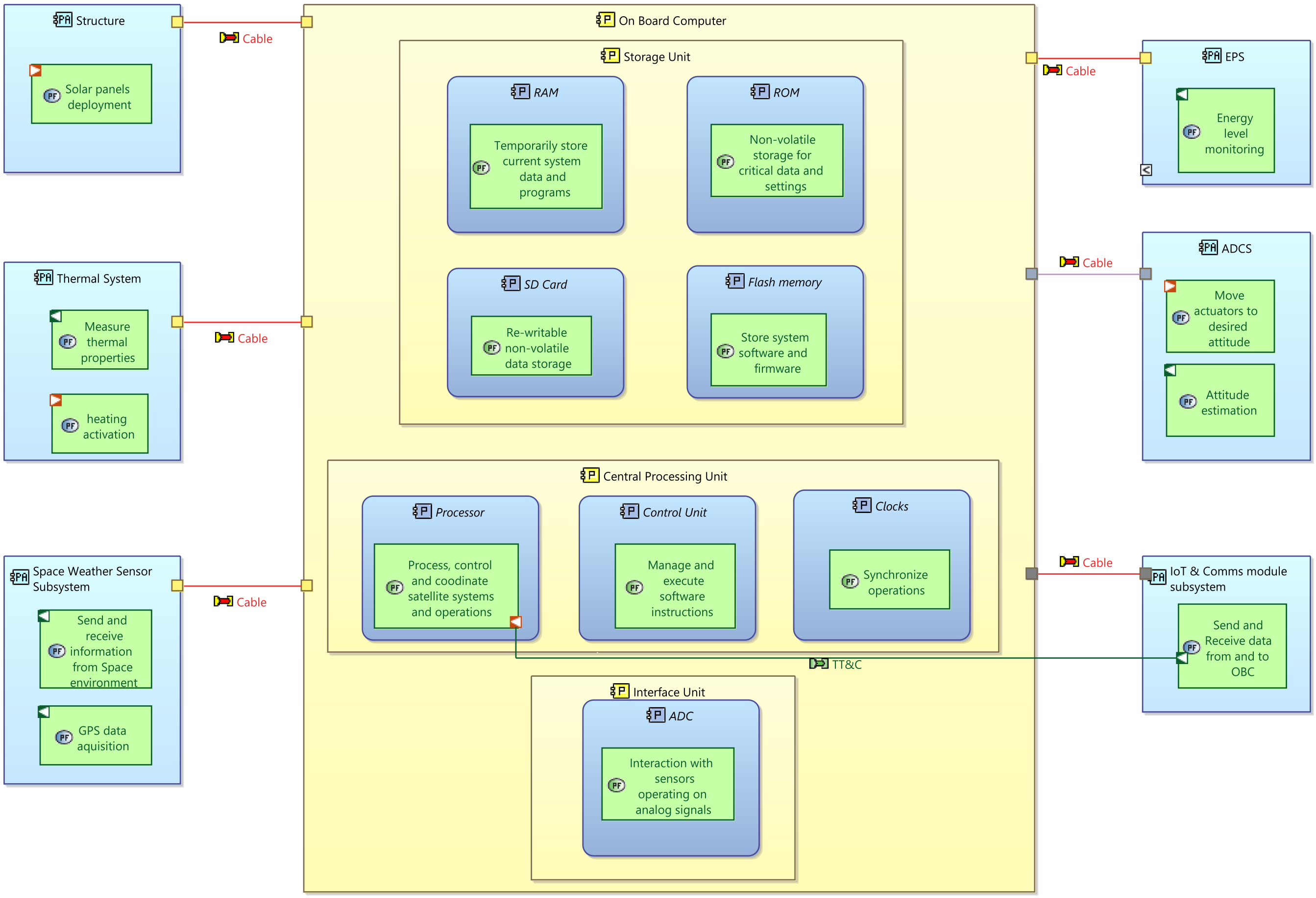


Fig. 14. ^{3}ColStar Physical Architecture of On-Board Computer SubSystem.

From here on, the diagrams shown are the diagrams of the satellite subsystems, the OBC (Fig 14), TT&C (Fig 15), ADCS (Fig 16), and EPS (Fig 17). The objective of this level is the same as for Logical Architecture, except that it defines the final architecture of the system, and how it must be carried out ("how the system will be built"). The physical diagrams are composed of the following elements:

- Behaviour Physical Component: Physical Component tasked with Physical Functions and therefore carrying out part of the behavior of the System (for example software component, data server, etc.).
- Physical Port : Non-oriented port that belongs to an Implementation Component (or Node). The structural port (Component Port), on the other hand, has to belong to a Behaviour

Component;

- Physical Link: Non-oriented material connection between Implementation Components (or Nodes). The Component Exchange remains a connection between Behaviour Components. A Physical Link allows one or several Component Exchanges to take place (for example USB cable, etc.).

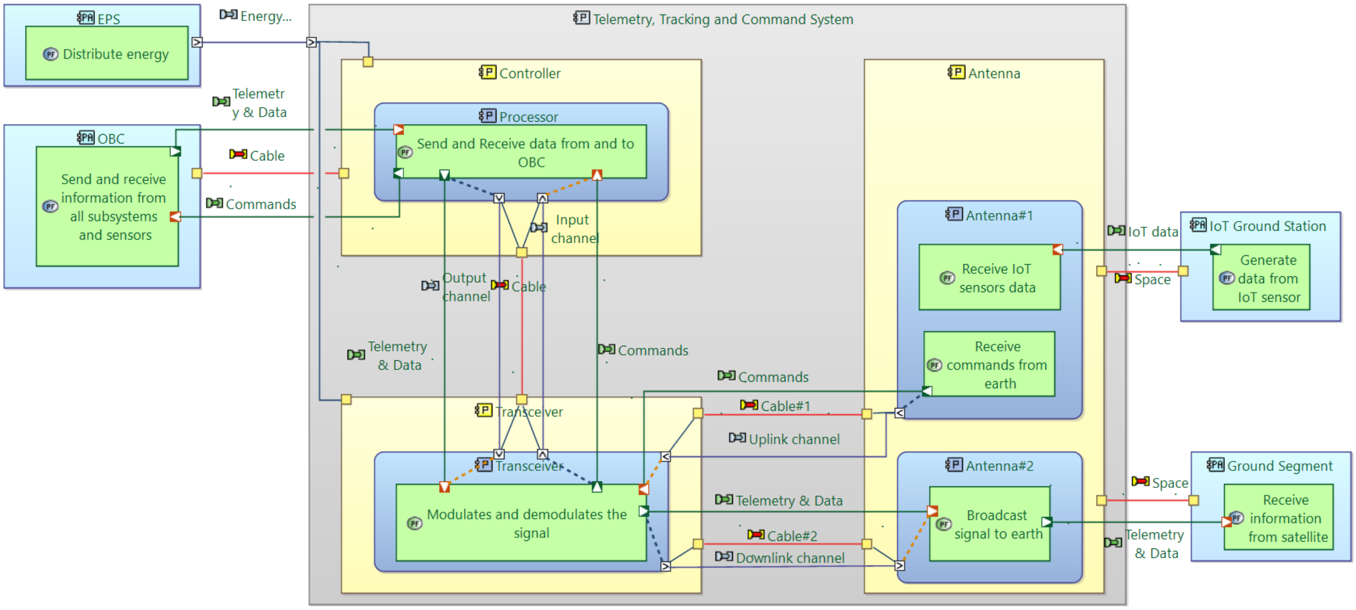


Fig. 15. [3]ColStar Physical Architecture of Communication SubSystem (TT&C).

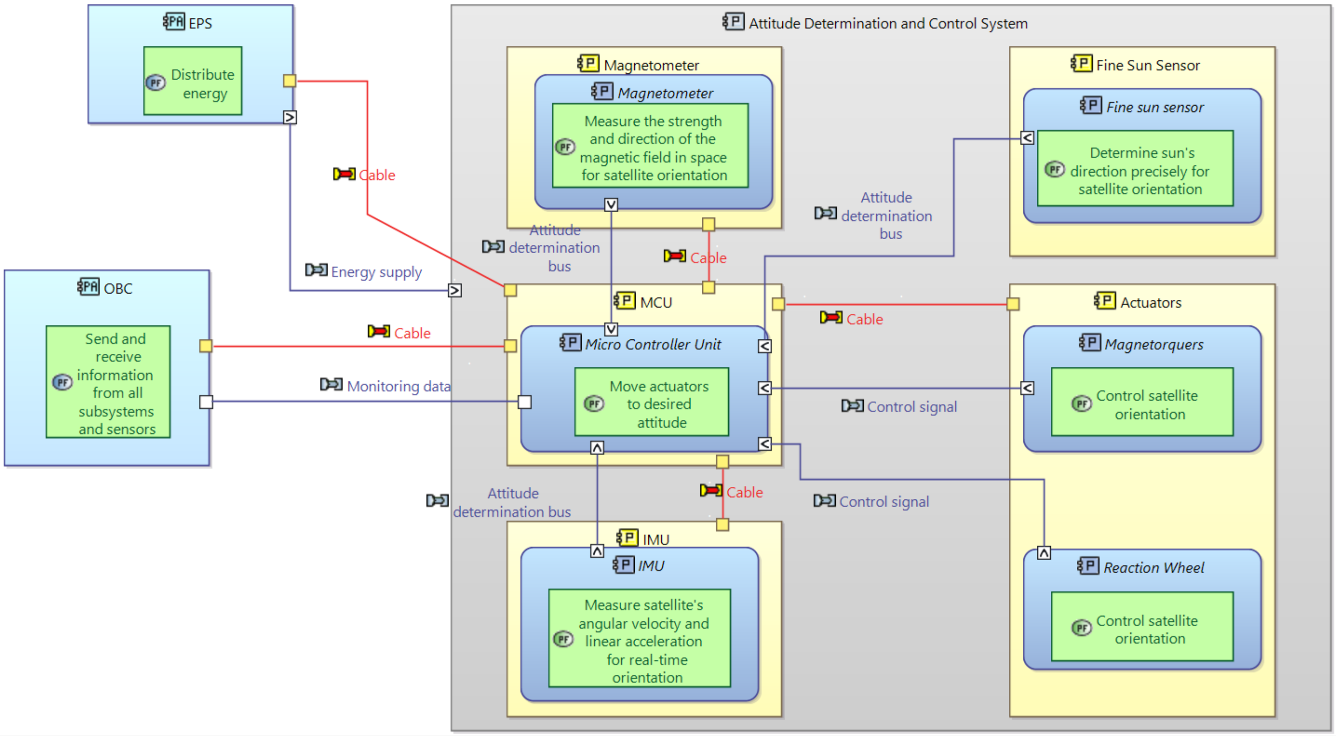


Fig. 16. ^{3}ColStar Physical Architecture of ADCS.

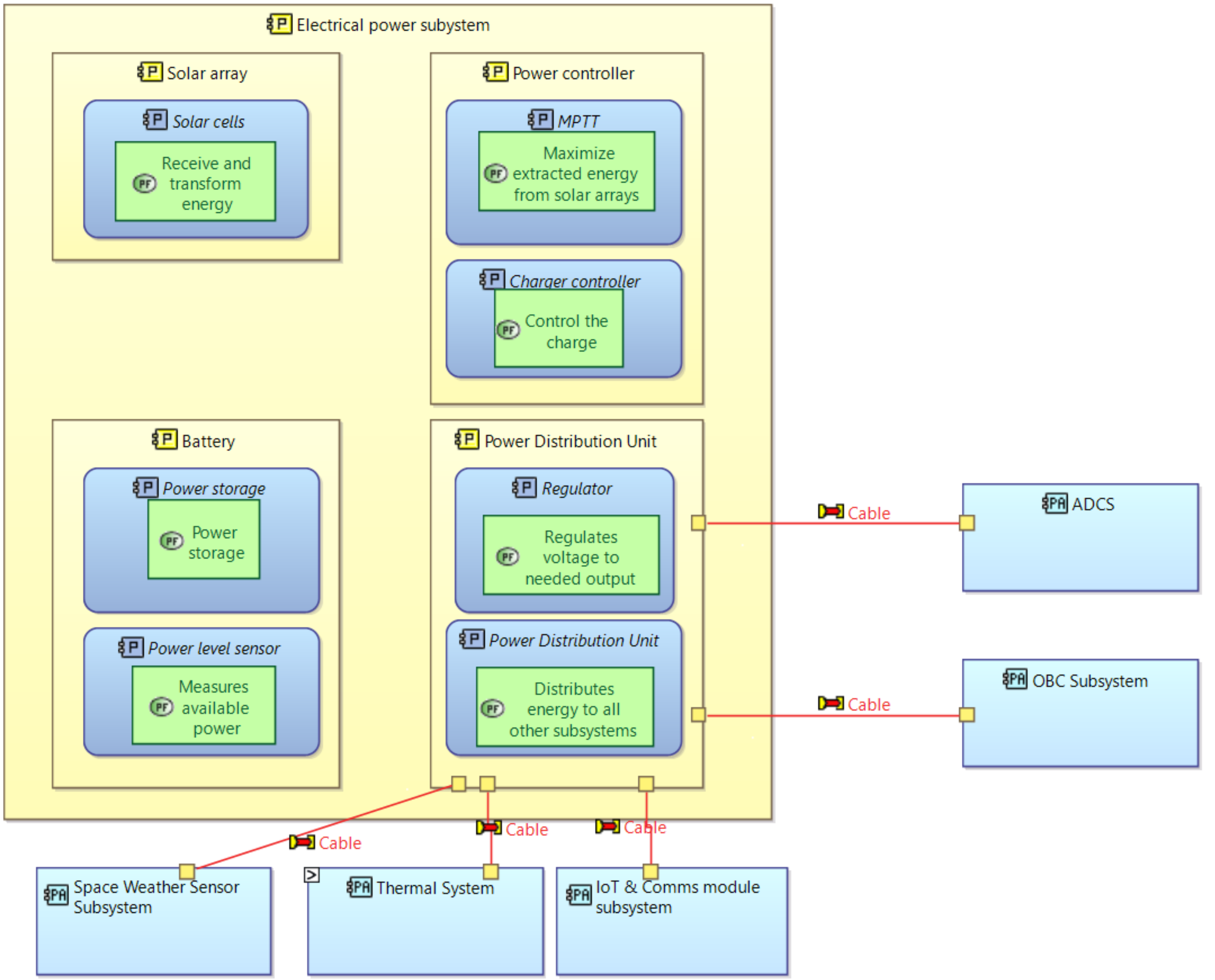


Fig. 17. ^{3}ColStar Physical Architecture of EPS.

F. Validation and Verification

The verification and validation process of a CubeSat mission involves several critical stages to ensure the mission's objectives are achievable and sustainable. At this stage of the project, only preliminary validations have been performed to assess various subsystems and mission requirements, focusing on orbital dynamics, ground station communication links, the electrical power system (EPS), and the Attitude Determination and Control System (ADCS) with a particular emphasis on the detumbling phase simulation.

The feasibility of the mission begins with a comprehensive simulation of the orbit of the satellite's orbit. This initial step is crucial to ascertain whether the CubeSat can maintain its prescribed trajectory while fulfilling mission objectives. Through orbital simulation, the team assesses the ability of the satellite to establish and maintain contact with ground stations. This simulation determines not only the feasibility of achieving consistent communication links but also the duration of these communication windows. Such analysis is vital for planning mission operations and ensuring that data transmission to and from the CubeSat is optimized.

Following the orbital assessment, the focus shifts to the CubeSat's communication subsystem. The verification of this subsystem is centered on ensuring reliable communication between the satellite and the ground stations. The validation process involves simulating the satellite's passage through various ground stations'coverage areas to confirm that the CubeSat can indeed make contact, as well as to estimate the quantity and duration of these communications. This step is critical for mission success, as it directly impacts data transmission, command and control, and overall mission operability.

To ensure the communications between the spacecraft and the different ground stations are correctly estimated, an orbital simulation of the mission profile has been executed by using an astrodynamics propagator (NASA GMAT) to calculate the number of contacts, duration, range, and mission lifetime (re-entry). Since the mission needs to be validated not only by running multiple orbital simulations based on the possible future date of launch but also against past KiboCube missions to compare real data against the computational simulation for 3ColStar, the mission selected is Moldovan's TUMnanoSAT[37], launched on July 15th, 2022. The official duration of the mission was around 200 days. In the case of 3ColStar cubesat, running the same mission launch date, the duration at the same altitude of EoL is 238 days, based on numerical integration errors from the propagator steps and computer models from atmospheric drag.

This difference in the perturbations is the result of a static model MSISE90 implemented in the simulation, the spherical ideal model for the solar radiation pressure and the J2 mathematical gravity perturbation.

Since the mission operations will be located in Colombia (Colombian Air Force SpOC), and due to its proximity to the Equator latitude, the number of contacts will be significantly reduced during mission operation. Therefore, an additional ground station was selected for the mission operations. This ground station is Barcelona's Observatorio de Montsec. Using this additional facility increases the amount of contacts to fulfill the mission objectives.The summary of the ground stations' contacts are shown in Table VI.

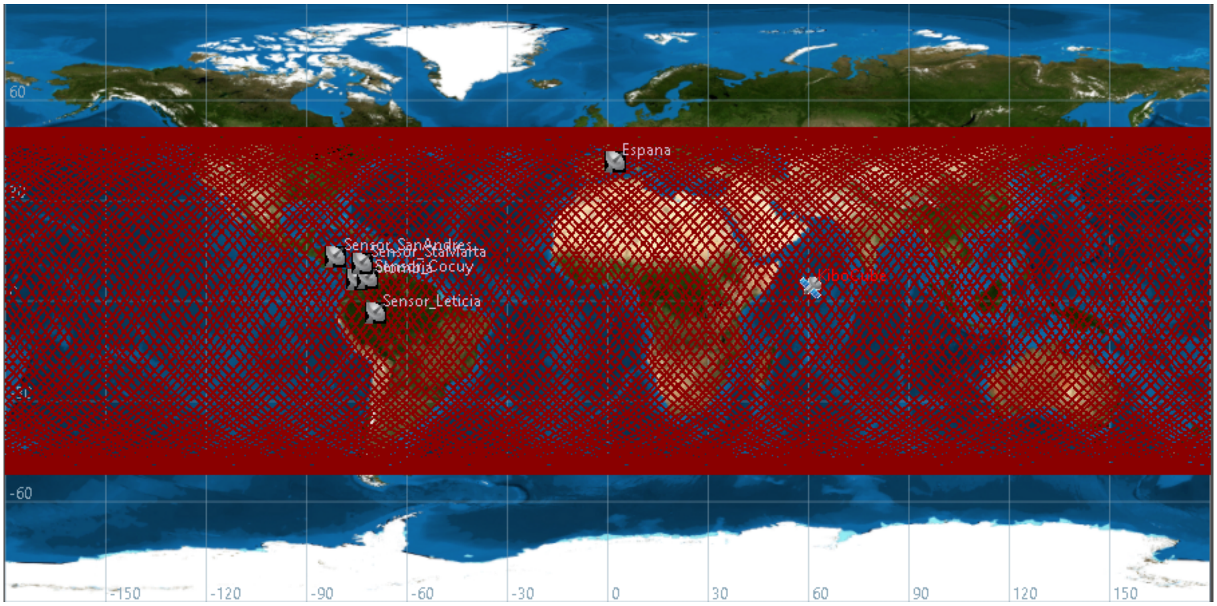


Fig. 18. ColstarCube Orbit passes

TABLE V

RE-ENTRY SIMULATION: RESULTS

Satellite initial altitude (ISS) (kms)	**Satellite final altitude (EOL) (kms)**	**Elapsed Days**
416.016	161.298	238.687

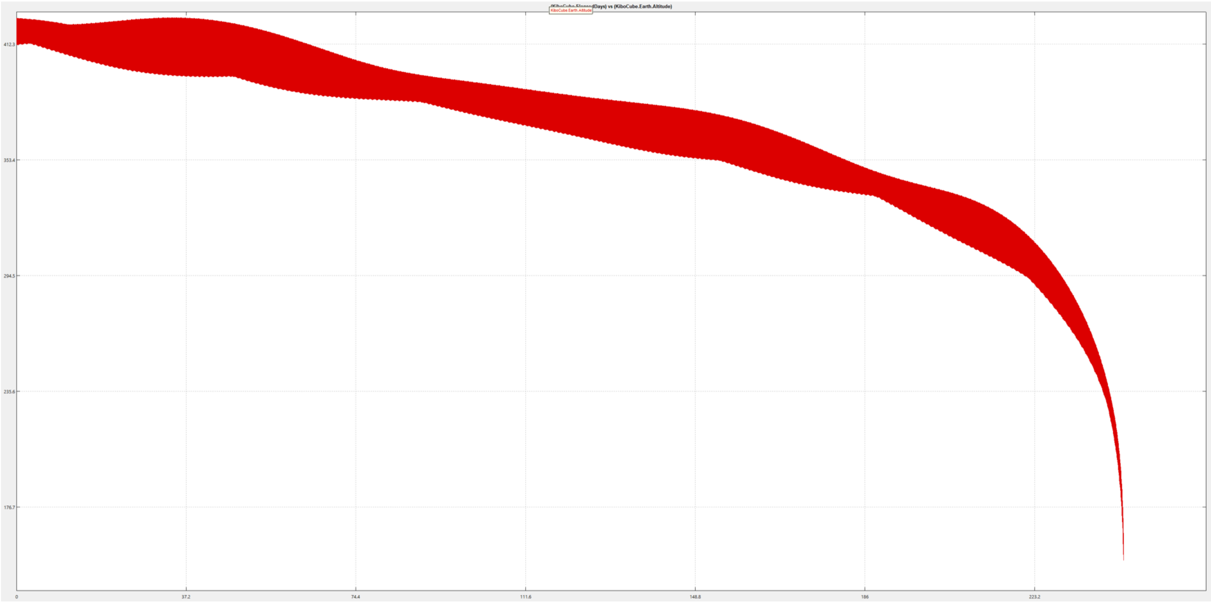


Fig. 19. Orbit decay of ^{3}ColStar Cube in GMAT.

TABLE VI

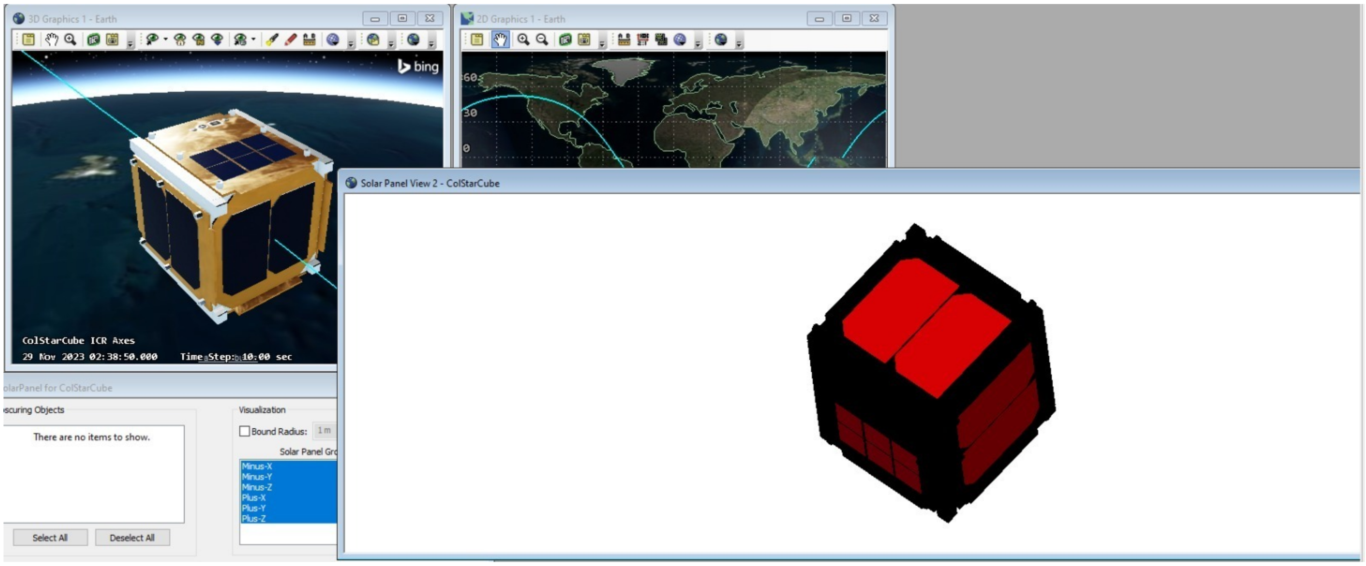
COLSTARCUBE CONTACTS WITH GS.

Ground Station	**Number of Contacts with Satellite**	**Average time of contact with Satellite (secs)**
Colombia FAC SPOC	572	284.745
Spain Montsec	1326	276.195

The EPS sizing is another pivotal aspect of the verification and validation process. Utilizing MATLAB, the team conducts simulations to determine the adequacy of the power supply for the mission's duration. This involves evaluating whether standard 1U CubeSat-sized solar panels suffice or if larger, deployable panels are necessary to meet the mission's energy requirements. The calculations also yield the size, capacity, and other characteristics of the battery. The simulations help in understanding the power consumption dynamics under different operational modes and conditions, ensuring that the CubeSat's power system can sustain its subsystems throughout the mission.

Initially, a 1U CubeSat equipped with six solar panels—one on each side—is considered for power generation. Each panel comprises two cells with a 1.2 Wp capacity and a 28 %

efficiency rate. Orbit simulation of the satellite indicates that the maximum energy generation per orbit is 2.52 Wh, assuming a 90 % efficiency rate and nadir orientation.



There are no items to show.

☐ Bound Radius: 1 m

Fig. 20. Orbit simulation in STK for EPS sizing.

TABLE VII

EPS SIMULATION RESULTS

Configuration	Mode	Energy Per Orbit [Wh]	Pmax (W)	E (Wh)90 %
Using Deployable Panels	Sun Pointing	6,47	6,84	5,82
	Nadir Pointing	2,72	3,54	2,45
No Deployable Panels	Sun Pointing	2,11	2,23	1,90
	Nadir Pointing	2,8	3,11	2,52

Given these findings, it becomes necessary to explore the market for a deployable solar panel system capable of exceeding the energy requirements of the satellite's operational modes and ensuring sufficient battery charging during eclipse passages. A configuration comprising three panels was chosen: one fixed to the CubeSat's surface and two extendable, allowing for simultaneous sun exposure across all panels. This setup, as per the simulation, generates 5.82 Wh of power, adequately covering the energy needs for any operational mode.

Regarding energy storage and considering the power needs during eclipse periods (Te), the operation mode with the highest energy consumption—Basic (Ba) mode, requiring 2.98 Wh—is selected. Factoring in a Depth of Discharge (DoD) of up to 35 % for the batteries and a 90 % energy transfer efficiency from the batteries to the CubeSat's systems, the calculated necessary battery capacity must be at least 9.46 Wh. Market research shows available batteries with comparable capacities, such as 9.6 Wh.

Lastly, for the ADCS subsystem, the detumbling stage of the satellite was simulated using MATLAB. The program used is an adaptation of the work found in [38].

The detumbling process is critical for stabilizing the CubeSat upon reaching orbit, transitioning it from an uncontrolled to a stable state. By simulating this phase, the team assesses the effectiveness of the system in achieving stabilization and estimates the time required to complete the detumbling process. This simulation ensures that the CubeSat can promptly begin its mission operations post-launch with a stable attitude control.

In figure 21 it is shown the time it takes for the satellite to stabilize after the deployment assuming initial values of 2 degrees per second. it is approximately 8000 seconds, which is about 2 hours or 2 orbits (Figure 24). using only the magnetorquers. The team found this and the current to magnetorquers (Figure 23) to be acceptable and within the desired values.

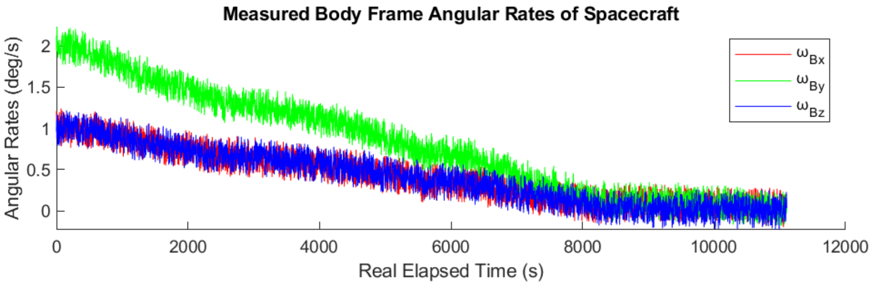


Fig. 21. Measured body frame angular rates of ^{3}ColStar

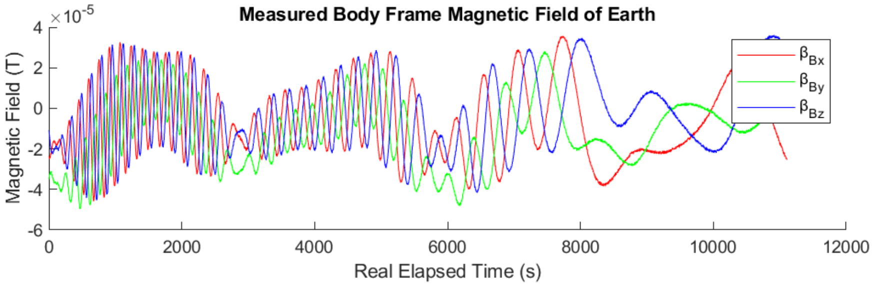


Fig. 22. Measured body frame magnetic field of earth.

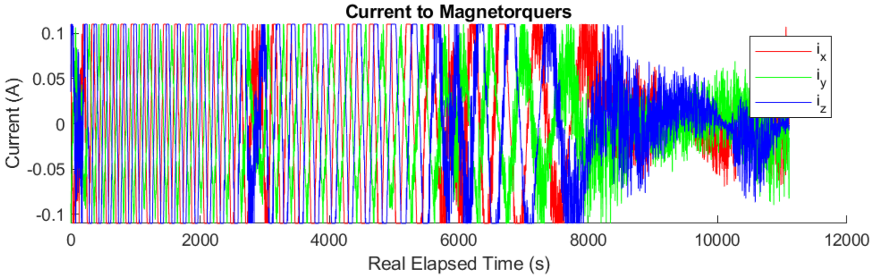


Fig. 23. Current to magnetorquers.

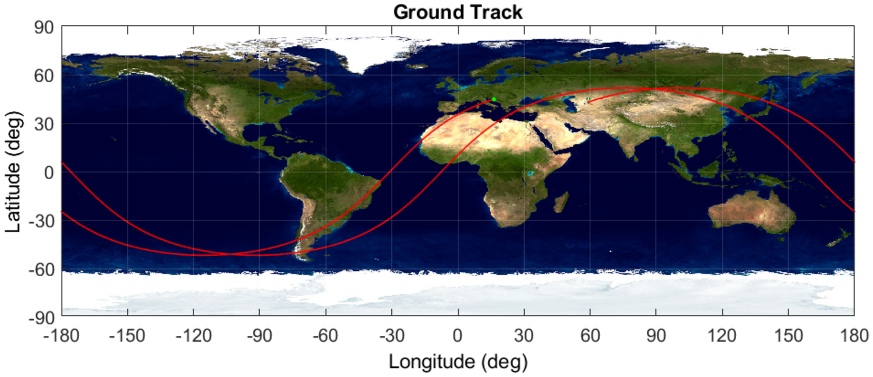


Fig. 24. Detumbling phase ground track.

G. Risk Analysis

Risk analysis in a satellite mission involves evaluating potential hazards and uncertainties that might affect the mission's success. This process includes identifying risks, assessing their impact and likelihood, and determining mitigation strategies. It is crucial because it helps ensure the safety, reliability, and success of the mission by proactively managing and minimizing risks, which can range from technical failures and cost overruns to environmental impacts and regulatory compliance. Effective risk management is essential for achieving mission objectives and safeguarding investments.

In reference to the specifications provided by JAXA and based on the experience documented in the NASA Risk Management Handbook [7], Safety Management at ESA the main risks that have been identified are considered, which can have critical consequences on the current phase of the CubeSat design. For risk assessment, the parameters of probability and impact or criticality scale proposed by the reference document will be used as a basis. After evaluating the risks and classifying them, corresponding measures will be proposed to either eliminate or mitigate the risk. In addition, the risks that it must be assumed due to the design of the nanosatellite for JAXA's knowledge and guidance will be determined.The systems analyzed are shown in Table VIII.

TABLE VIII

ABBREVIATIONS USED FOR THIS RISK ANALYSIS

Abbreviations	**Related**
ADCS	Attitude Determination and Control System
EPS	Electrical Power System
OBC	On Board Computer
COM	Communications
ST	Structure
TH	Thermal
TM	Team
SCH	Schedule
PL	Payload

1) Likelihood or probability or occurence: Probability of occurrence refers to the estimate of how likely a specific event, situation, or risk is to occur. The failure estimate was made based on the statistics carried out by NASA on partial or total failure of CubeSat-type satellite missions from 2000 to 2016.

TABLE IX

LIKELIHOOD SCALE

5	Maximum	Certain or almost certain to occur, will occur at least once the chance is 1 to 1.
4	High	Will occur frequently, the chance is between 1 to 1 and 1 to 10.
3	Medium	Will occur sometimes, the chance is between 1 to 10 and 1 to 100.
2	Low	Will seldom occur, the chance is between 1 to 100 and 1 to 1000.
1	Minimum	Will almost never occur; the chance is less than 1 to 1000.

2) Risk severity: In the context of risk analysis for a project like a CubeSat, "Risk Severity" or "Impact" refers to the extent of harm, damage, or negative consequences that could result if the risk were to materialize. It's a measure of the potential effect on the project's objectives, performance, schedule, cost, or technical outcomes. Impact is often categorized into levels such as low, medium, and high, based on criteria specific to the project.

Risk impact parameters to consider for evaluation and analysis are shown in Table X.

TABLE X

SEVERITY SCALE

5	Maximum	Unacceptable, no alternatives exist.
4	High	Major reduction, but workaround available.
3	Medium	Moderate reduction, bur workaround available.
2	Low	Moderate reduction, some approach retained.
1	Minimum	Minimal or no Impact.

TABLE XI

RISK QUANTIFICATION MATRIX

5		5	10	15	20	25
4		4	8	12	16	20
3		3	6	9	12	15

2	2	4	6	8	10
1	1	2	3	4	5
Severity/Likelihood	**1**	**2**	**3**	**4**	**5**

TABLE XII

RISK QUANTIFICATION SCALE

Risk Quantification	Magnitude	
>20	Maximum	Maximum disruption of project plan, maximum threat to project success, implement new process or change baseline plan
15-19	High	High disruption of project plan, large threat to project success, implement new process of change baseline plan.
10-14	Medium	Some disruption of project plan, some threat to project success, aggressively manage, consider alternative process.
5-9	Low	Little disruption of project plan, little threat to project success some management actions necessary
<5	Minimum	No disruption of project plan, no threat to project success, current approach is sufficient

TABLE XIII

RISK QUANTIFICATION MATRIX

5	Low	Medium	High	Maximum	Maximum
4	Minimum	Low	Medium	High	Maximum
3	Minimum	Low	Low	Medium	High
2	Minimum	Minimum	Low	Low	Medium
1	Minimum	Minimum	Minimum	Minimum	Low
Severity/Likelihood	**1**	**2**	**3**	**4**	**5**

TABLE XIV

IDENTIFIED RISKS FOR THE MISSION

Code	**Risk Name**	**S**	**L**	**SxL**	**Description**	**Mitigation**
EPS 1	Temperature Space variations	3	3	Low	Possible impact on payload sensors and potential damage the nanosatellite batteries	Analysis of the optimal location in the Cubesat. Thermal analysis
EPS 2	Battery failure	4	3	Medium	Satellite would lack electrical power.	Subsystem disconnection to operating with solar panels power supply.
EPS 3	Battery degradation	4	1	Minimum	Less electrical power available for distribution in the satellite.	Automatic reloop programming.

EPS 4	Solar panel Deployment Failure	3	2	Low	Power supply from fixed solar panels.	Change of operating mode to optimize electrical energy distribution.
EPS 5	NO solar radiation resistance	4	1	Minimum	Impact on electrical components due to solar radiation.	analysis of electrical components and optimal internal distribution design.
EPS 6	Subsystem interphase failure	4	1	Minimum	Inadequate distribution of electrical power in subsystems may cause degradation in their operation.	Guidelines from the International Space Station (ISS
EPS 7	Operation modes Failures (SU, SS, Nominal,Survival)	4	1	Minimum	Abnormal development of the mission	Monitoring of battery status and energy consumption. Mode operations test on ground.
EPS 8	Converter failite	5	1	Low	Power failures in subsystems can damage them of affect their performance.	Operational time in accordance with the non-significant lifespan.

EPS 9	Electromagnetic compatibility	3	1	Minimum	ISS damage due to compatibility	Magnetic field requirement measurement.
OBC 1	MCU (Microcontroller unit) Failure	4	1	Minimum	Security vulnerabilities can be exploited, compromising mission control and integrity.	Operational checking and tolerance verification through tests during development.
OBC2	Overload of EPS towards OCB	5	2	Low	Loss of data, interrupted control, and possibly mission failure.	Contemplate overload protection in the EPS (Electric power subsystem) architecture
OBC3	Excessive electrical power consumption in operation	4	1	Minimum	Security vulnerabilities can be exploited, compromising mission control and integrity.	Detailed analysis of power consumption and distribution in the nanosatellite.
OBC4	Solar radiation disturbances or solar storms	5	2	Medium	Loss of data, interrupted control, and possibly mission failure.	Assume risk, pre-evalauation of affectation level on low orbit.

OBC 5	Interface failure for nanosatellite control (software)	3	3	Low	Security vulnerabilities can be exploited, compromising mission control and integrity.	Remote control to check the status of the On-Board Computer (OBC), automatic contingency mode operation to stabilize the internal operation of the nanosatellite.
OBC 6	Cybersecurity	4	1	Low	Loss of data, interrupted control, and possibly mission failure.	Encryption of codes to enhance data integrity security levels of the nanosatellite
OBC 7	Intermittent communications	3	4	Medium	Security vulnerabilities can be exploited, compromising mission control and integrity.	Autonomy for task control without connection.

COM 1	Electromagnetic interference with other satellites	3	1	Minimum	Disruptions in communication signals and potentially impact the functionality of neighboring satellites.	Direct signal, incorporation of signal encryption. Radiofrequency laboratory tests to verify communication functionality in the aeroponic chamber.
COM 2	Up/Downlink interference comm failure	5	4	High	Errors in the communication link, whether in the transmission (uplink) or reception (downlink), may result from issues in the satellite's hardware or software, affecting data transfer reliability.	Ground station partnerships and deployment analysis for an enhanced network link. Implementation of synchronization and timing optimization in the software.

COM 3	Synchronization and Timing Problems.	5	1	Low	Disruptions in the coordinated operation of different components within the system. Tasks that depend on precise timing may not be executed as intended.	Implementation of synchronization and timing optimization of software.
COM 4	Loss of the primary power source.	5	1	Low	Failure to provide electrical power, possible loss of communications	Activation of modes to prioritize power supply activation when required. Reduction of the main sensor data frame to decrease the required power consumption by reducing transmission time.

COM 5	Software issues interface	3	3	Low	Interface affectation ground station and the use of LoRa (Long Range) communication technology.	Satellite antenna acquisition and development of the antenna for IoT. Centralized configuration of access to the central satellite from the ground station to isolate telemetry transmission from one transmitting via IoT.
COM 6	There is no frequency assignment	3	2	Low	ITU and IARU do not provide a frequency band to transmit	Evaluation of changing frequency and paying for using a frequency band
COM 7	Unexpected RF emissions	3	1	Minimum	Effects on ISS equipment.	Initial design thinking to be in allowable range. Test to determine maximum output power.

ST 1	Insufficient Structural Resistance	4	3	Medium	Deformation or damage to the satellite structure can lead to catasthophic results.	Virtual simulated Shall comply jmx-2012694
ST 2	Impact from Micrometeoroids or Space Debris	5	2	Medium	Structural impact or failure of the nanosatellite can compromise its overall integrity.	Simulation through software such as DAS (Debris Assessment Software) and DRAMA (Debris Risk Assessment and Mitigation Analysis) is conducted to mitigate the risk of collision with other satellites, meteoroids, or space debris in low Earth orbit.

ST 3	Vibration Resistance during Launch	4	1	Minimum	Structural overall integrity.	Vibration tests are performed on the structure following established guidelines, and assembly procedures undergo verification and validation.
ST 4	Assembly Issues	4	1	Minimum	It weakens the structure and has the potential to jeopardize the functionality and protection of other components.	Quality inspections are carried out using tools and techniques, and a lifespan analysis is conducted.
ST 5	Material Fatigue	4	1	Minimum	Structural fatigue failure caused by temperature decay and radiation degradation.	Certified and characterized materials are selected in accordance with the specified requirements.
ST 6	Incorrect deployment	2	3	Low	Collision with other operational satellites or spacecraft.	Deployment analysis. Adherence to deployment procedures

TM 1	Team availability	4	2	Low	Schedule Delays	Multiple institutions committed to the project.
SCH 1	Software and subsystem development time longer than expected.	4	3	Medium	Schedule Delays	Interface software and development basis experience in FACSAT Nanosatellites
THI	Thermal variation and heat distribution problems	5	2	Medium	Affectation of nanosatellite integrity and operation due to thermal variation	Precise characterization of thermal properties, design optimization for adequate heat distribution
TH2	Errors in thermal modeling	5	2	Medium	Incorrect predictions of thermal behavior in space	Regular review and validation of the thermal model to improve analysis accuracy.

ADCS 1	Attitude sensor failure	4	3	Medium	Attitude sensors such as magnetometers or gyroscopes may fail or provide incorrect readings due to hardware failures or electromagnetic interference.	Implement sensor redundancy, conduct periodic calibrations, and develop robust sensor fusion algorithms to identify and disregard erroneous data.
ADCS 2	Attitude control actuator failure	4	3	Medium	Actuators like reaction wheels or thrusters may fail due to mechanical or electrical issues, affecting the satellite's ability to alter its orientation.	Use redundant actuators, conduct preventive maintenance based on telemetry data analysis, and develop emergency attitude control procedures in case of failures.

ADCS 3	Attitude control software errors	4	2	Low	Software errors or failures in control algorithms can lead to unexpected behaviors in attitude control.	Employ safe software development practices, conduct exhaustive testing including simulations and flight tests where possible, and maintain an in-orbit software update mechanism.
ADCS 4	Electromagnetic Interference (EMI)	3	3	Low	EMI can interfere with the proper functioning of ADCS components, including sensors and actuators.	Design the CubeSat with adequate EMI protection, including the use of shielding materials and implementing filters on power and data lines.

V. PROJECT MANAGEMENT IMPLEMENTATION

A. Work Breakdown Structure

For a better understanding of the necessary steps to meet the objectives of the project within the defined schedule, the whole work since the kick-off until the launch of the satellite were divided into work packages. The logic used on ^{3}ColStar KiboCUBE Colombia Team is represented on the next Figure, which divides the work firstly for segments, then for main tasks, and inside the most complex main tasks are even divided into small tasks. The objective is decomposing a complex work into small manageable tasks or activities, which could be assigned to a member of the team or a sub-team. Also, it represents the lower level work packages division adopted by the project, specifically for the subsystems that will be developed for the satellite. This lower level was only assumed for segment "Development" being the most complex segment and with the highest human resources allocation needed for meeting the deadlines.

The WBS of the system is divided on the first level by its phases, for a better comprehension of the tasks by phase necessary to a successful project. Figure 25 shows the WBS generated for the preliminary mission of this project.

Figures 28,27 and 28 show the proposed Work Breakdown Structure of the following stages of the project. It is important to highlight that the WBS allows activities to be related to the resources necessary to carry out these activities and the respective deliverables. The WBS for all stages were designed following an agile mindset with taking into account the proposed scrum framework found in subsection B and the NASA Project Life Cycle [8] and its main deliverables.

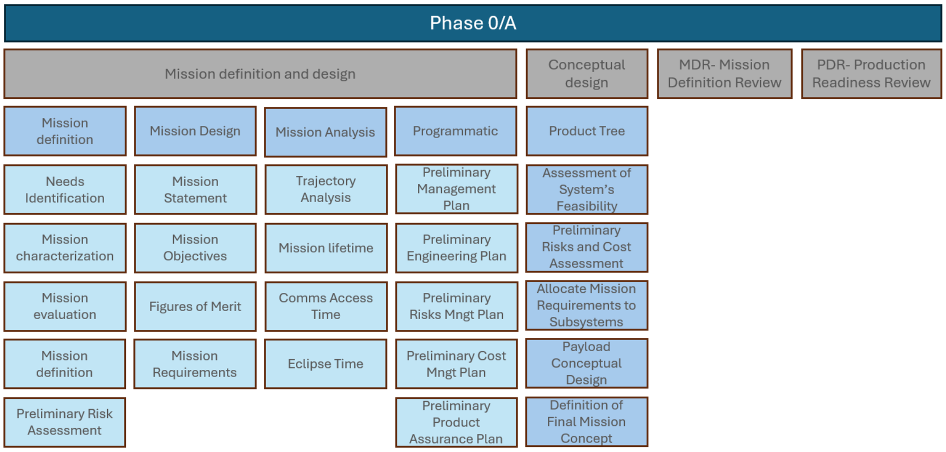


Fig. 25. Work Breakdown structure of Phase A.

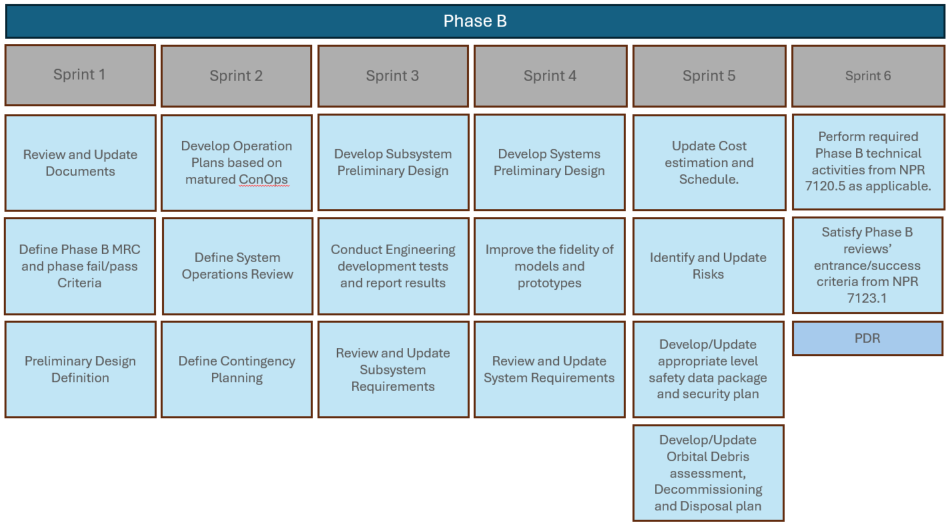


Fig. 26. Work Breakdown structure of Phase B.

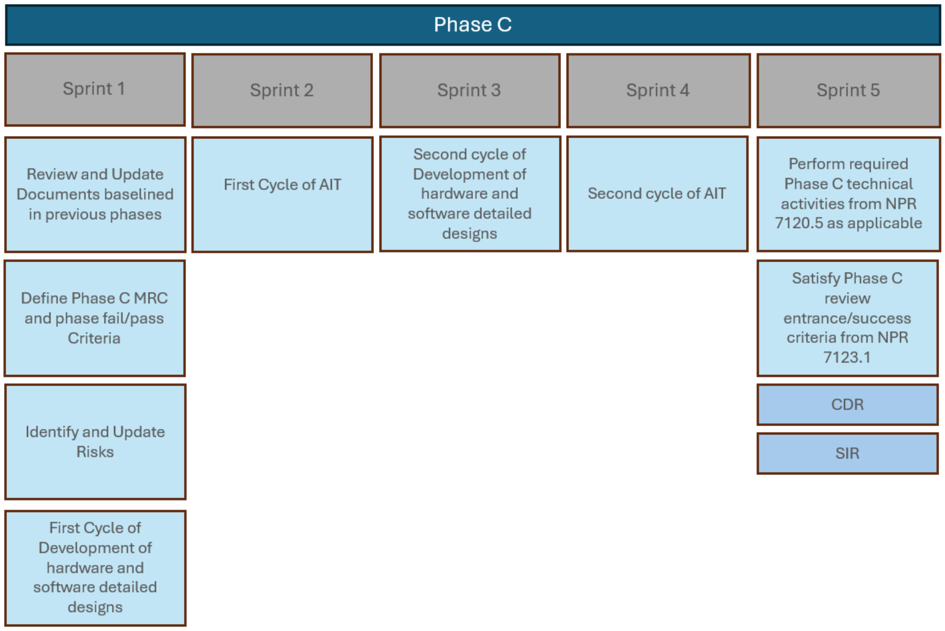


Fig. 27. Work Breakdown structure of Phase C.

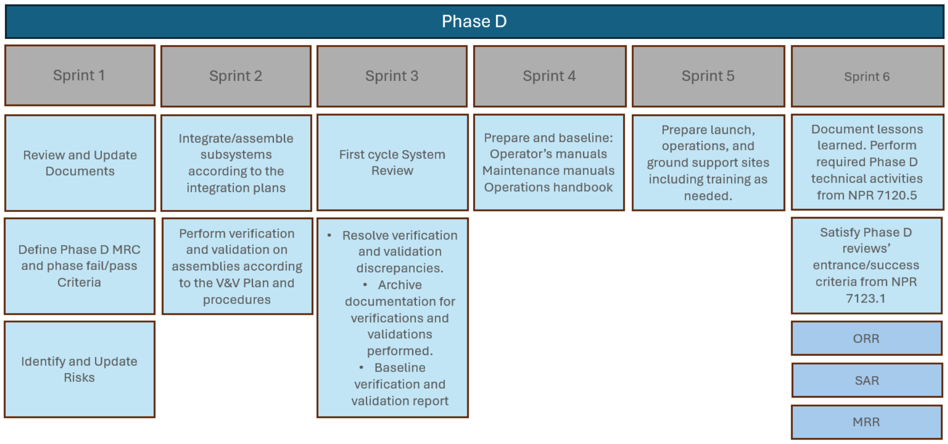


Fig. 28. Work Breakdown structure of Phase D.

B. Modified Agile methodology

Since the Manifesto for Agile Software Development was introduced in 2001, agile practices have transformed how software teams create products. The Manifesto outlines a series of core values and principles aimed at enhancing software development [2]. It has led to various methodologies and frameworks like Scrum, Kanban, and Lean, along with other terms and techniques [39].

Hardware and software development involve distinct developmental tasks. While Scrum, an Agile methodology commonly applied to software development, might not initially appear suitable for hardware development, the apparent disparities mainly revolve around the nature and order of deliverables, rather than fundamental constraints on the process itself.

Some differences of hardware development with software development are [40]:

- Software is more malleable (easier to change) than hardware. The cost of change is much higher for hardware than for software.
- Specialized hardware parts may take significantly longer to acquire compared to software.
- Software products develop over time with successive releases, involving the addition of new features and the refinement of existing ones. In contrast, hardware products primarily comprise physical components that cannot be easily altered after manufacturing like software. They cannot gain new capabilities through simple modifications.
- Architectural decisions heavily influence the design of a hardware product, needing a greater upfront investment in architectural planning due to the high cost of making changes later, unlike in software products.

Although agile adoption is relatively new for hardware, there are already some proposed frameworks, such as Modified Agile for Hardware Development (MAHD) [41]. In which there is a section of upfront work called MAHD on ramp, and then move on to sprints. Likewise, MAHD is not based on incremental development, but on iterative design and early validation. Another difference is that this framework uses a focus matrix to prioritize product attributes. There are several aerospace based development projects in which the scrum methodology has been applied [42]. Other works include manufacturing and launching a cubesat [43][44][45]. As a result of work such as this, a slightly deeper understanding has been gained as to whether agile methodologies are compatible with space development, which practices can be adopted and which cannot, given the unique characteristics of hardware development and key aspects of the space sector such as a focus on safety, reliability and risk aversed [46], aspects not commonly found in software development.

During the proposal structuring and preliminary design stage of the ^{3}ColStar project, the development team adopted a modified Scrum methodology, tailored to suit their specific needs. The team was organized into subsystems, including Guidance, Navigation, and Control (GNC), Thermal, On-Board Computer (OBC), Systems Engineering (SE), among others, as illustrated in Figure 11. This approach differed slightly from the original Scrum framework depicted in Figure 29. Sprints were conducted on a weekly basis, with each team maintaining its own product backlog. Additionally, daily scrum meetings were held within each team to ensure effective communication and progress tracking. The framework is displayed in Figure 30 and two parallel processes for two different teams are shown. This is because each sub-team or subsystem operates with its own independent backlog and follows a distinct scrum process. Unlike the typical software development approach where there is a single backlog for the whole team.This and the focus on MBSE allowed working in an agile way and implementing changes quickly. For the next phases of the project, which already involve hardware, manufacturing, integration and testing, it is proposed a modified scrum methodology that can adapt correctly to the characteristics of work in these next phases, based on the experiences and lessons learned in the referenced projects show that a hybrid approach can be successful (for instance, the MBSE for Ariane 6 [26]). In this proposal, there are differences in key aspects such as:

- Upfront work: although it is not possible to know all requirements exactly at the beginning of a project, it is not possible to eliminate the need to have an estimate of requirements at the beginning of the project.
- Epics: a fundamental difference between a software development work team and a space mission work team is that in the latter, if there are defined roles, such as the control, electrical or thermal engineer, the project must be divided into different epics, which correspond to the subsystems.
- Product backlog: In this, instead of using user stories, which usually come out of the conversation between the product owner and the customer, the requirements estimated at the beginning of the project would be used. These requirements can be broken down and translated into tasks, which would end up being the user stories. It is also necessary to

have different product backlogs for each epic, that is, a product backlog for the tasks and needs of each subsystem.

- Sprint: Commonly, the duration of sprints in software development lasts one or two weeks, since software allows rapid prototyping and release to market. However, some works such as [44] show that, for hardware, it is better to have slightly longer sprints, 3 or 4 weeks or even more. Therefore, it is proposed that the duration of sprints initially be 4 weeks.
- Releases: In software development, at the end of each sprint it is usual to find the release of the new software features, and this is functional, however, this may not be very realistic in applying it to the manufacture of a cubesat, since every month should have a prototype of a cubesat with the new modifications and improvements. This is costly and inefficient, so it is proposed that by the end of each sprint, instead of having a complete CubeSat prototype, the increment in the final product should be to verify and validate some of the tasks and requirements that have been previously chosen for the sprint backlog of each subsystem or epic.

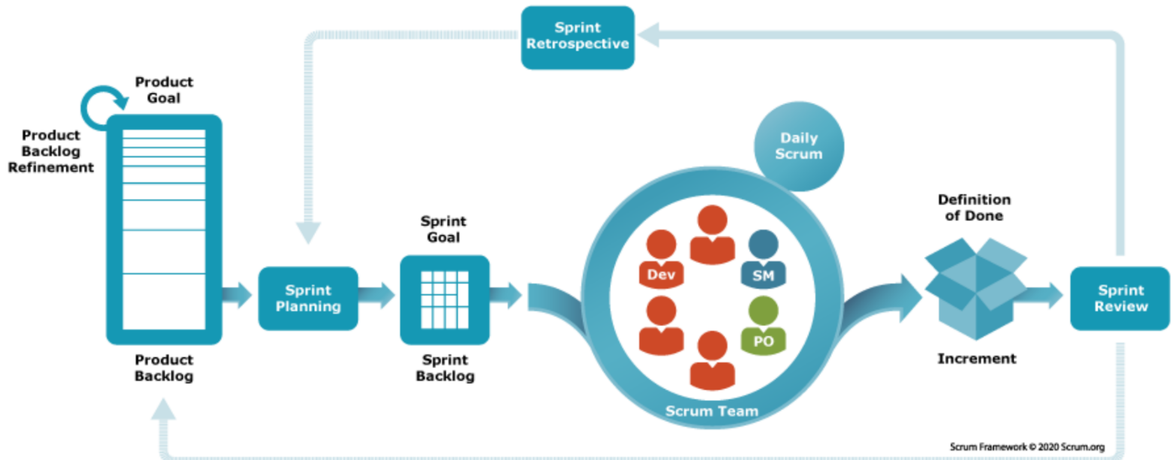


Fig. 29. Scrum methodology in software development [5] [6].Image taken from [7]

C. Design and Development Schedule

For the design and development schedule, the software tool named Jira is used. Jira, developed by Atlassian, is a powerful project management and issue tracking software wi-

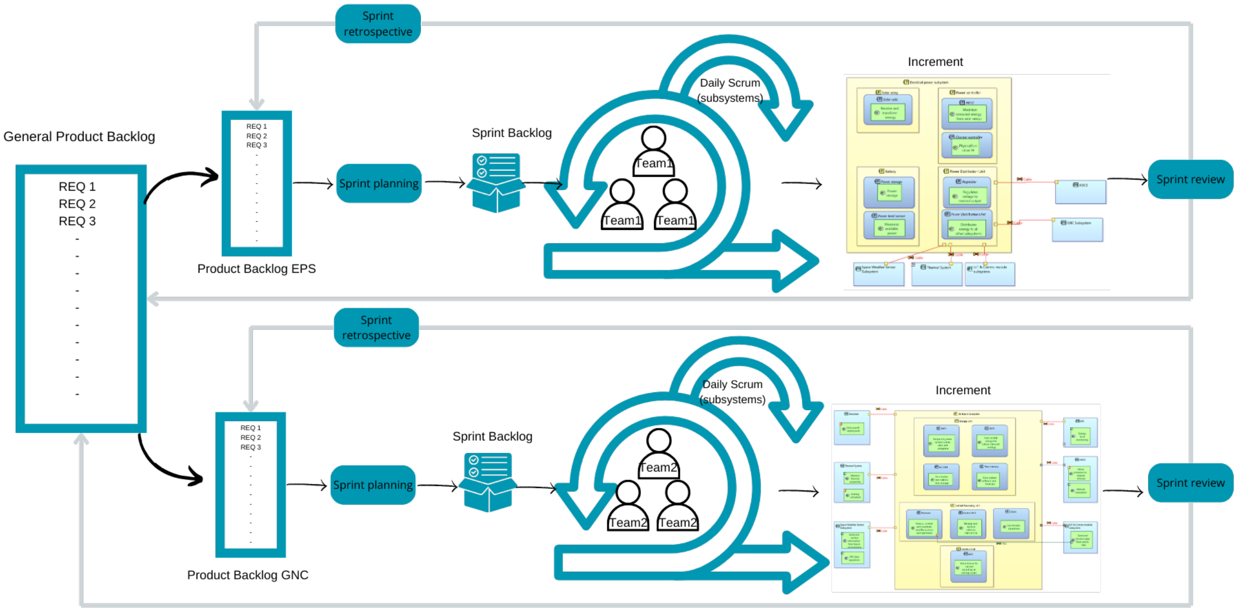


Fig. 30. Scrum methodology being used in the ^{3}ColStar KiboCUBE.

dely used across various industries, including Aerospace, for project management. Initially designed as a bug and issue tracker, Jira has evolved into a comprehensive project management tool that supports Agile methodologies, such as Scrum and Kanban, making it highly versatile for managing complex projects [47].

For the design of the schedule, the scrum methodology and agile philosophy described in section B are taken into account, as well as the NASA project life cycle for a robotic mission shown in Figure 31.

NASA life cycle phases are used to help plan and manage all major aerospace system developments. Everything that should be done to accomplish a project is divided into distinct phases, separated by control gates that have to be passed to proceed.

The project life-cycle phases of formulation and implementation are divided into incremental pieces. This allows the development team to access their progress, estimate system and project performance, plan the next phase and allows decision-makers to assess management and technical progress.

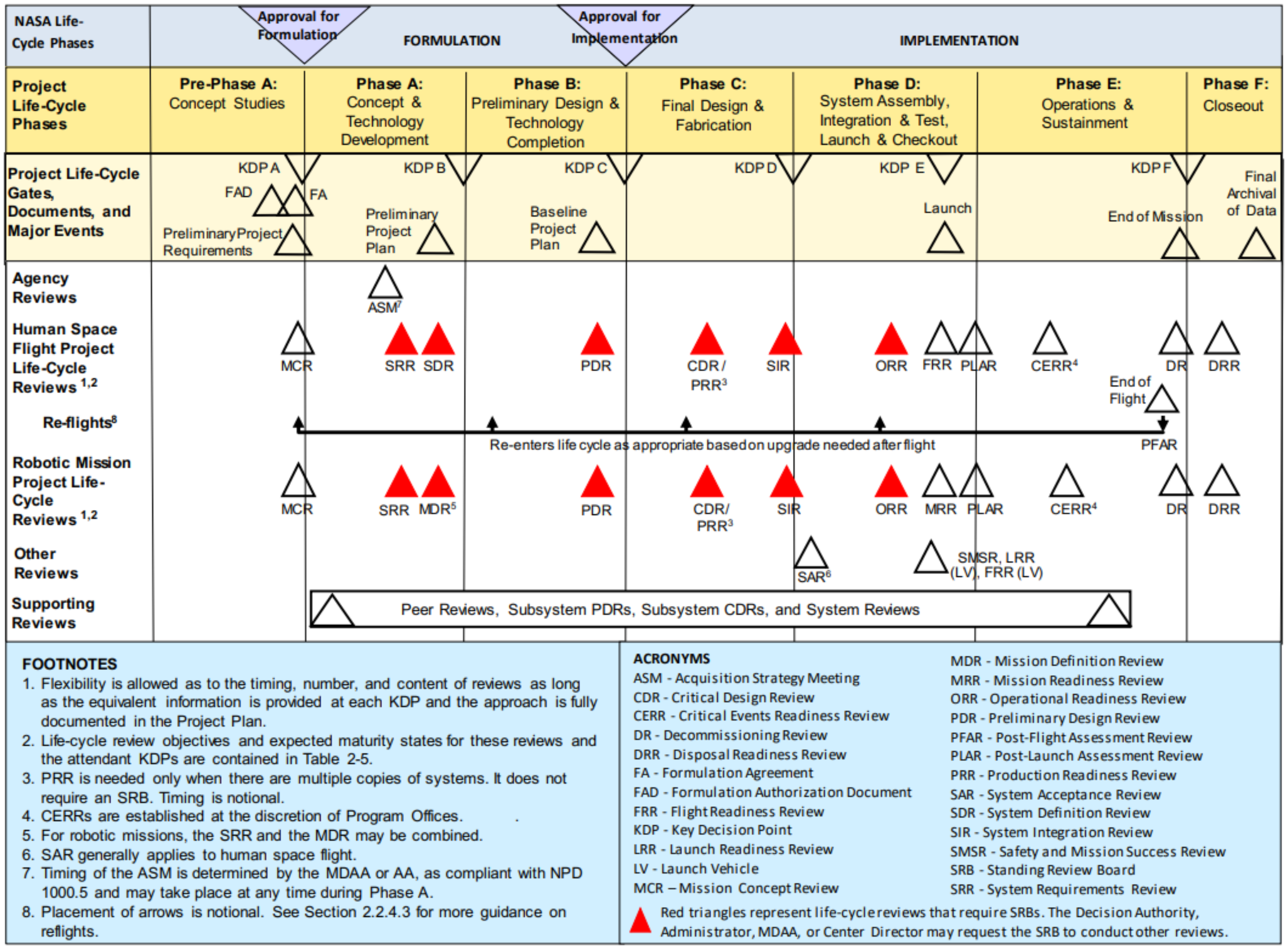


Fig. 31. NASA Project LifeCycle [8]

Detailed Sprint Plan Each sprint cycle will include planning, execution, daily stand-ups, review, and retrospective meetings. The cycle focuses on iterative improvements, stakeholder engagement, and ensuring project milestones are met efficiently.

Continuous Activities Throughout the sprints, continuous activities such as stakeholder engagement, backlog grooming, risk management, and quality assurance are paramount for project success.

The incremental work divided in sprints for the next phases of the project are as follows:

Sprint 1: Documentation and Monitoring

- Review and update documents baselined in previous phases.
- Monitor progress against plans.
- Define Phase B MRC and phase fail/pass criteria.

Sprint 2: Operations Planning

- Develop operations plans based on matured ConOps.
  → Define system operations, review, and access and contingency planning.

Sprint 3: Subsystem Preliminary Design

- Develop the subsystem preliminary design.
  → Conduct engineering development tests as needed and report results.

Sprint 4: System Preliminary Design and Prototyping

- Develop the System preliminary design.
- Improve the fidelity of models and prototypes.

Sprint 5: Cost, Risk, and Safety Planning

- Update cost range estimate and schedule data.
- Identify and update risks.
- Develop appropriate level safety data package and security plan.
- Develop/Update preliminary plans.
  → Orbital Debris Assessment.
  → Decommissioning Plan.
  → Disposal Plan.

Sprint 6: Technical Activities and Phase B Review

- Perform required Phase B technical activities from NPR 7120.5 as applicable.
- Satisfy Phase B reviews' entrance/success criteria from NPR 7123.1.

Phase C

Sprint 7: Phase C Initial Cycle

- Review and update documents baselined in previous phases.
- Monitor progress against plans.
- Define Phase C MRC and phase fail/pass criteria.
- Identify and update risks.
- First Cycle of Development of hardware and software detailed designs.

→ Add remaining lower-level design specifications to the system architecture.
→ Perform development testing at the component or subsystem level.
→ Fully document final design and develop data package.

Sprint 8: First Cycle of AIT

- First cycle of AIT.

→ Interface definitions.
→ Manufacturing and assembly.
→ Subsystem verification and validation.

Sprint 9: Second Development Cycle

- Second cycle of Development of hardware and software detailed designs at the subsystem level.

→ Add remaining lower-level design specifications to the system architecture.
→ Perform development testing at the component or subsystem level.
→ Fully document final design and develop data package.

Sprint 10: Second Cycle of AIT

- Second cycle of AIT.

→ Interface definitions.
→ Manufacturing and assembly.
→ Testing at the component or subsystems.
→ Subsystem verification and validation according to the V&V Plan and procedures.

Phase D

Sprint 12: Update Documents and Risk Management

- Update documents developed and baselined in previous phases.
- Monitor project progress against plans.
- Define Phase D MRC and phase fail/pass criteria.
- Identify and update risks.

Sprint 13: Integration and Validation

- Integrate/assemble subsystems according to the integration plans.
- Perform verification and validation on assemblies according to the V&V Plan and procedures.
  - → Perform system qualification verifications, including environmental verifications.
  - → Perform system acceptance verifications and validation(s) (e.g., end-to-end tests encompassing all elements, i.e., space element, ground system, data processing system).
  - → Assess and approve verification and validation results.

Sprint 14: First Cycle System Review

- First cycle System Review.
  - → Resolve verification and validation discrepancies.
  - → Archive documentation for verifications and validations performed.
  - → Baseline verification and validation report.

Sprint 15: Preparation and Baseline

- Prepare and baseline:
  - → Operator's manuals.
  - → Maintenance manuals.
  - → Operations handbook.

Sprint 16: Launch and Operations Preparation

- Prepare launch, operations, and ground support sites including training as needed.
  - → Train initial system operators and maintainers.
  - → Train on contingency planning.
  - → Confirm telemetry validation and ground data processing.
  - → Confirm system and support elements are ready for flight.
  - → Provide support to the launch and checkout of the system.
  - → Perform planned on-orbit operational verification(s) and validation(s).

Sprint 17: Documentation and Review

- Document lessons learned. Perform required Phase D technical activities from NPR 7120.5.
- Satisfy Phase D reviews' entrance/success criteria from NPR 7123.1.

Figure 34 shows a graphic way to describe the relationship between the main deriverables and the epics of phases of the project in a agile or cyclic context, in which each arm of the spiral is an epic or phase (phase B,C,D). On the other hand, the gantt chart in Figure 32 shows all the sprints since the beginning of the project.

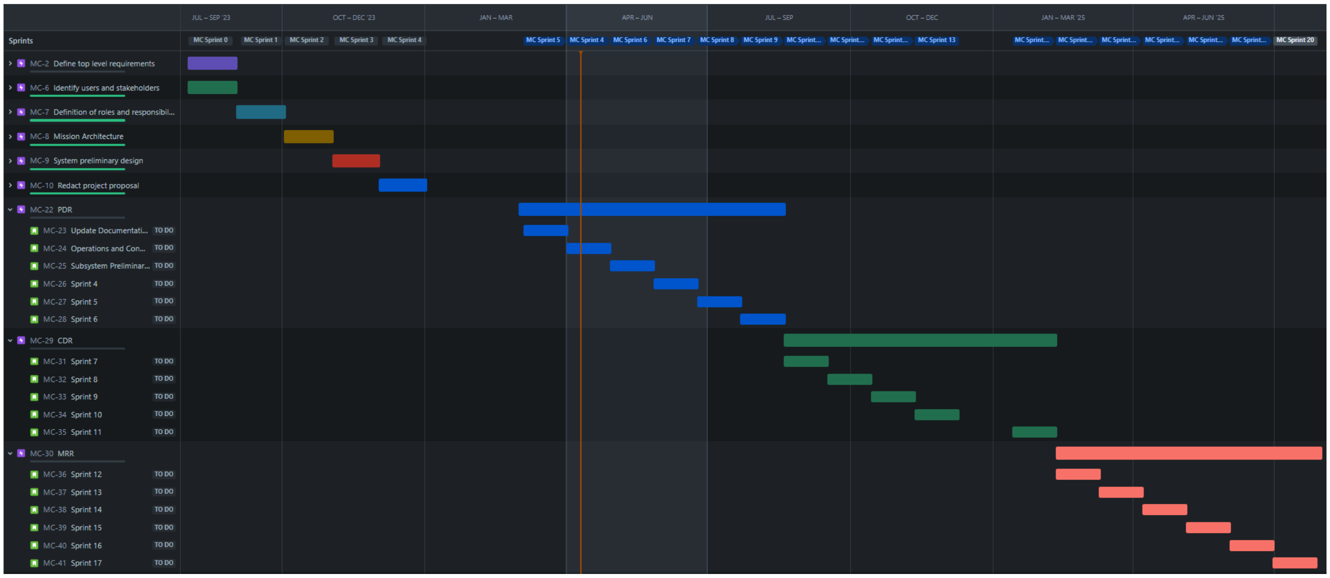


Fig. 32. Gantt Chart in Jira for the phases A,B,C,D

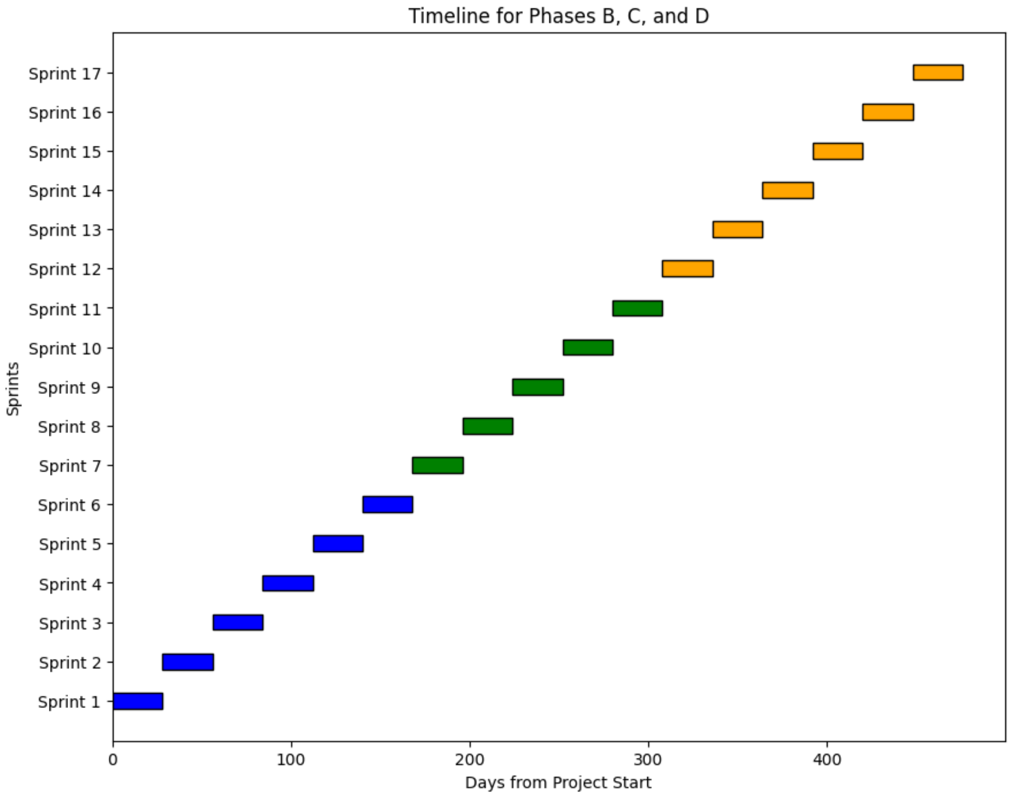


Fig. 33. Sprints scheduled for the upcoming phases of the proposed Scrum methodology

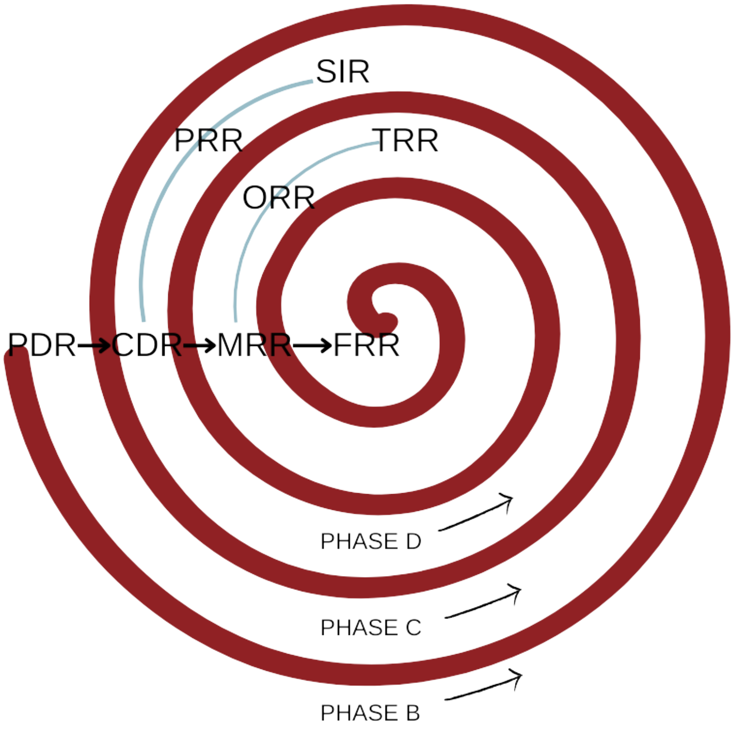


Fig. 34. The NASA LifeCycle represented in a spiral, in which each arm represents and epich in the scrum methodology.

D. Costs estimation

For the cost estimation, it was taken as reference the commercial value of the components and the commercial value of cubesats of the same size and similar mission profile.It is important to clarify that this is a preliminary estimate and that prices may change with a more rigorous estimate.

TABLE XV
COST ESTIMATION

Item	**Cost (USD)**	**Observations**
Scientific Team	$250,000	Members Team

Structure	$80,000	In-House/In-Kind
EPS		
OBC		
ADCS (RW- 3 MGT)	$5,000	In-House/In-Kind
Thermal/Solar Pannels	$50,000	In-House/In-Kind
COMMS	$34,000	In House
Weather Sensor	$75,000	Purchase/In-House
IoT	$29,810	In House
Fine sun sensor	$3,000	In-House/In-Kind
Electronic Components	$5,000	Purchase
Structure and thermal analysis software	$20,000	Purchase/In-Kind
Adninistrative process	$20,000	In-House
Others	$10,000	Purchase
Testing facilities	$40,000	In-House/In-Kind
Ground Segment	$100,000	In-House/In-Kind
Travel and per diems (Japan Working activities)	$16,000	
STEM Program	$20,000	In-House/In-Kind
Launch	-	
Engineering model (flatsat)	$20,000	In-House/In-Kind
Back-up structure		In-House/In-Kind
Flight model	$60,000	In-House/In-Kind
SUBTOTAL (without Colombian taxes)	$837,810	
RETIFUENT 20 % FINANCIAL EXPENSES	$117,562	
FINANCIAL EXPENSES 14 %	$133,752	
OPERATING EXPENSES 5 %	$54,456	

VI. CONCLUSIONS

In conclusion, the application of the Arcadia method for the Preliminary Design of the ^{3}ColStar mission enabled the completion of this process within a remarkable timeframe of 4 months, facilitated by weekly general sprints. This timeframe compares favorably with the recommendations provided in the Space project management- Project planning and implementation (ECSS-M-ST-10C Rev. 1) [48], and NASA Procedural Requirements (NPR) [49], which suggest a duration of 1 to 6 months for similar processes. The utilization of this methodology in other 1U CubeSat missions has demonstrated improved conceptualization, enhanced communication between subsystems, and technical precision, thereby minimizing errors in mission development.

Furthermore, the incorporation of Capella into the approach provided a means to achieve technical consistency through digitalized processes, aligning with the principles of agile methodology. The implementation of Capella facilitated the creation of a Shared system model with multiple views, connected to discipline models, thereby formalizing aspects of systems engineering through model-based systems engineering (MBSE). It's crucial to note that MBSE does not replace traditional systems engineering but rather supplements it with rigorous methods and tools, ensuring coherency within the model and managing technical consistency across various architectural documents.

Moreover,The ^{3}ColStar mission's VV efforts in the preliminary phase were comprehensive, covering crucial aspects from orbital dynamics and communication systems to the satellite's electrical power supply and attitude control systems. By using advanced simulation tools like GMAT for orbit simulation and MATLAB for subsystem testing, the team could identify and address potential issues early in the development phase. This proactive approach reduces the risk of mission failure and enhances overall mission reliability.

One significant aspect of the VV process was its iterative nature. As issues were identified, solutions were implemented and tested iteratively, allowing the team to refine the satellite's design continuously. This iterative process is crucial in complex engineering projects where initial designs may not meet all operational requirements due to the unpredictable nature of the space environment.

In summary, the integration of agile systems engineering principles, Arcadia methodology, and Capella tools has significantly streamlined the preliminary design phase of the ^{3}ColStar satellite mission. This approach not only expedites the design process but also enhances the accuracy, communication, and efficiency of mission development, setting a precedent for future CubeSat missions and beyond.

VII. RECOMMENDATIONS

It is recommended to continue enhancing the integration of agile methodologies and Model-Based Systems Engineering (MBSE). This includes providing more training to ensure all team members are proficient in these approaches, which will help streamline the satellite construction process. Additionally, establishing robust systems for knowledge sharing is crucial due to the high turnover rates associated with an academic project, especially in university settings. Implementing comprehensive documentation practices, mentorship programs, and digital repositories can greatly enhance continuity and knowledge transfer among team members.

Moreover, developing advanced risk management frameworks that utilize predictive analytics and real-time data is essential for proactively managing potential delays and budget overruns.

Expanding training on agile project management across all team levels will enhance adaptability and responsiveness to project dynamics, satellite projects can change rapidly, all team members should be trained in agile methods. This includes workshops and practice sessions that mimic real project conditions.

Also, strengthening collaborations with the private sector will not only enhance learning opportunities but also improve resource sharing and foster innovation for the team. It is also advisable to focus on sustainability practices in satellite design and operation to minimize environmental impact and ensure safe decommissioning of satellites after their mission ends.

Furthermore, adapting agile tools specifically for hardware projects can address the unique challenges posed by the longer lead times and higher costs associated with hardware modifications. After the completion of satellite missions, conducting thorough post-mission analyses can provide valuable insights and lessons that can be applied to future projects.

REFERENCES

[1] "^{3}ColStar KiboCUBE Colombia Team. 8th Round CubeSat Mission Application," 2024*, internal Document.

[2] "Manifesto for Agile Software Development." [Online]. Available: https://agilemanifesto.org/

[3] "Endurosat 1U Cubesat platform," Aug 2023. [Online]. Available: https://www.endurosat.com/cubesat-store/cubesat-platforms/1u-cubesat-platform/

[4] P. Kovář, P. Puričer, E. Benton, and J. Mikeš, "Demonstration of the capability of 1U CubeSat for measurement of the Energy Spectrum on leo," *Electronics*, vol. 11, no. 20, p. 3390, Oct 2022.

[5] J. Sutherland and K. Schwaber, "The scrum papers," *Nuts, bolts and origins of an Agile process*, 2007.

[6] V. Hema, S. Thota, S. N. Kumar, C. Padmaja, C. B. R. Krishna, and K. Mahender, "Scrum: an effective software development agile tool," in *IOP Conference Series: Materials Science and Engineering*, vol. 981, no. 2. IOP Publishing, 2020, p. 022060.

[7] "What is Scrum?" [Online]. Available: https://www.scrum.org/resources/what-scrum-module

[8] S. J. Kapurch, *NASA systems engineering handbook*. Diane Publishing, 2010.

[9] K. M. Adams, P. T. Hester, J. M. Bradley, T. J. Meyers, and C. B. Keating, "Systems theory as the foundation for understanding systems," *Systems Engineering*, vol. 17, pp. 112–123, 3 2014.

[10] P. Zimmerman, T. Gilbert, and F. Salvatore, "Digital Engineering transformation across the Department of Defense," *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, vol. 16, no. 4, p. 325–338, Dec 2017.

[11] J. Geraldi, H. Maylor, and T. Williams, “Now, let’s make it really complex (complicated),” *International Journal of Operations amp; Production Management*, vol. 31, no. 9, p. 966–990, Aug 2011.

[12] E. C. Honour, “A historical perspective on systems engineering,” *Systems Engineering*, vol. 21, no. 3, p. 148–151, Mar 2018.

[13] E. H. Livermore, “Integrating Agile Systems Engineering and Project Management in Small Satellites Development,” Ph.D. dissertation, NTNU, 2022.

[14] S. A. Sheard and A. Mostashari, “Principles of Complex Systems for Systems Engineering,” *Systems Engineering*, vol. 12, no. 4, p. 295–311, Nov 2009.

[15] S. C. Spangelo, D. Kaslow, C. Delp, B. Cole, L. Anderson, E. Fosse, B. S. Gilbert, L. Hartman, T. Kahn, and J. Cutler, “Applying model based systems engineering (MBSE) to a standard CubeSat,” *2012 IEEE Aerospace Conference*, Mar 2012.

[16] S. Shoshany-Tavory, E. Peleg, A. Zonnenshain, and G. Yudilevitch, “Model-based-systems-engineering for conceptual design: An integrative approach,” *Systems Engineering*, vol. 26, no. 6, p. 783–799, Apr 2023.

[17] J. Gregory, L. Berthoud, T. Tryfonas, A. Rossignol, and L. Faure, “The long and winding road: MBSE adoption for Functional Avionics of spacecraft,” *Journal of Systems and Software*, vol. 160, p. 110453, Feb 2020.

[18] E. Honoré-Livermore and C. Haskins, “Model-based systems engineering for CubeSat FMECA,” *Recent Trends and Advances in Model Based Systems Engineering*, p. 529–540, 2022.

[19] “United Nations/Japan Cooperation Programme on CubeSat Deployment from the International Space Station (ISS) Japanese Experiment Module ”KiboCUBE”.” [Online]. Available: https://www.unoosa.org/oosa/en/ourwork/access2space4all/KiboCUBE/KiboCUBE_Index.html

[20] “MiniPIX TPX3 SPACE.” [Online]. Available: https://advacam.com/camera/minipix-space/

[21] Alexander Kossiakoff, Samuel J. Seymour, David A. Flanigan, Steven M. Biemer, *SYSTEMS ENGINEERING PRINCIPLES AND PRACTICE*, andrew p. sage. ed. JOHN WILEY SONS, INC., 2020.

[22] Z. Li, F. Faheem, and S. Husung, “Collaborative Model-Based Systems Engineering Using Dataspaces and SysML v2,” *MDPI Systems*, vol. 12, 2024.

[23] C. M. Crumbly, “Systems Engineering Technology: Closing the MBSE Modeling Gap through Community College,” *AIAA SCITECH 2024 Forum*, 2024.

[24] R. Eramo, M. Nolletti, L. Pomante, L. Pasquale, and D. Pascucci, “Model-driven engineering for simulation models interoperability: A case study in space industry,” *Software: Practice and Experience*, 2024.

[25] B. Ravikanti, H. Ali, M. G. Balchanos, E. Harrison, and D. N. Mavris, “MBSE-Enabled System Verification of Unmanned Aerial System (UAS) Noise Certification,” in *AIAA SCITECH 2024 Forum*, 2024, p. 1627.

[26] H. Heibrok, A. Donner, M. Edelhäuser, and T. Franz, “Open-source MBSE workflow for automated spacecraft thermal analyses from a system model IAC-22, D1, 4B, 2, x70017,” *Acta Astronautica*, vol. 214, pp. 253–260, 2024.

[27] J. L. Fernandez and C. Hernandez, *Practical model-based systems engineering.* Artech House, 2019.

[28] C. Grise, “Model Based System Engineering for Ariane 6.” 2019.

[29] “Arcadia Method and Capella.” [Online]. Available: https://mbse-capella.org/arcadia.html

[30] M. Kirshner and R. Valerdi, “Integrating Model-Based Systems and Digital Engineering for Crewed Mars Mission Planning,” *Journal of Aerospace Information Systems*, vol. 19, no. 10, pp. 668–676, 2022.

[31] A. Boydston, P. Feiler, S. Vestal, and B. Lewis, “Architecture centric virtual integration process (ACVIP): A key component of the DoD digital engineering strategy,” in *22nd Annual Systems and Mission Engineering Conference*, 2019.

[32] B. R. Williams, “Applied Agile Digital Mission Engineering for Cislunar Space Domain Awareness,” 2022.

[33] S. Lee, A. Hutputanasin, A. Toorian, W. Lan, R. Munakata, J. Carnahan, D. Pignatelli, and A. Mehrparvar, “CubeSat design specification, rev. 13: The CubeSat program,” *San Luis Obispo, California Polytechnic State University*, 2014.

[34] A. Poghosyan and A. Golkar, “CubeSat evolution: Analyzing CubeSat capabilities for conducting science missions,” *Progress in Aerospace Sciences*, vol. 88, pp. 59–83, 2017.

[35] T. Matsumura, Y. Akiba, J. Borrill, Y. Chinone, M. Dobbs, H. Fuke, A. Ghribi, M. Hasegawa, K. Hattori, M. Hattori *et al.*, “Mission design of LiteBIRD,” *Journal of Low Temperature Physics*, vol. 176, pp. 733–740, 2014.

[36] G. Tyc, J. Tulip, D. Schulten, M. Krischke, and M. Oxfort, “The RapidEye mission design,” *Acta Astronautica*, vol. 56, no. 1-2, pp. 213–219, 2005.

[37] Technical University of Moldova, “TUMnanoSAT,” https://nanosat.utm.md/, 2023, accessed: 2024-04-13.

[38] E. Christhuraj, “CubeSat Detumbling Simulator,” 2023, accessed: 2023-04-13. [Online]. Available: https://echristhuraj.github.io/CubeSatDetumblingSimulator/

[39] R. D. Estrada-Esponda, M. López-Benítez, G. Matturro, and J. C. Osorio-Gómez, “Selection of software agile practices using Analytic hierarchy process,” *Heliyon*, vol. 10, no. 1, 2024.

[40] "Agile Processes for Hardware Development." [Online]. Available: https://www.cprime.com/wp-content/uploads/woocommerce$_$uploads/2015/10/Agile-Processes-for-Hardware-Development-cPrime.pdf

[41] "An Introduction to Modified Agile for Hardware Development Framework: The Benefits of Agile, the Needs of Physical Product Development." [Online]. Available: https://mahdframework.com/

[42] J. Furuhjelm, J. Segertoft, J. Justice, and J. Sutherland, "Owning the Sky with Agile: Building a Jet Fighter Faster, Cheaper, Better with Scrum," *Global Scrum Gathering, San Diego, CA*, 2017.

[43] B. Segret, A. Semery, J. Vannitsen, B. Mosser, J.-J. Miau, J.-C. Juang, and F. Deleflie, "The Paving Stones: initial feed-back on an attempt to apply the AGILE principles for the development of a CubeSat space mission to Mars," in *Modeling, Systems Engineering, and Project Management for Astronomy VI*, vol. 9150. SPIE, 2014, pp. 322–331.

[44] N. Garzaniti, S. Briatore, C. Fortin, and A. Golkar, "Effectiveness of the scrum methodology for agile development of space hardware," in *2019 IEEE Aerospace Conference*. IEEE, 2019, pp. 1–8.

[45] K. Wortman, B. Duncan, and E. Melin, "Agile methodology for spacecraft ground software development: A cultural shift," in *2017 IEEE Aerospace Conference*. IEEE, 2017, pp. 1–8.

[46] S. E. Carpenter and A. Dagnino, "Is Agile too fragile for space-based systems engineeering?" in *2014 IEEE international conference on space mission challenges for information technology*. IEEE, 2014, pp. 38–45.

[47] Atlassian, "JIRA: Software for Team Planning and Project Management," https://www.atlassian.com/es/software/jira, accessed: 2024-04-13.

[48] E. Secretariat, "ECSS-M-ST-10C Rev1,Project planning, and implementation." 2009, european Cooperation for Space Standardization (ECSS).

[49] T. L. Osborne, *NASA Space Flight Program and Project Management Handbook*. National Aeronautics and Space Administration (NASA), 2022.