

UNIVERSIDAD NACIONAL DEL ALTIPLANO

**FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA,
ELECTRONICA Y SISTEMAS**

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



**“SOFTWARE DE TELECONTROL Y
TELEMETRÍA PARA EL PROCESO DE
MEDICIÓN Y CONTROL DEL NANOSATÉLITE
WARA”**

TESIS

PRESENTADO POR:

RUDOL KATARI HUALLPA LLANOS

PARA OPTAR EL TITULO PROFESIONAL DE:

INGENIERO DE SISTEMAS

Puno – Perú

2014

UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRONICA Y SISTEMAS

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

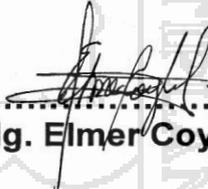
**“SOFTWARE DE TELECONTROL Y TELEMETRÍA
PARA EL PROCESO DE MEDICIÓN Y CONTROL
DEL NANOSATÉLITE WARÁ”**

TESIS PRESENTADA POR:

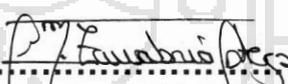
RUDOL KATARI HUALLPA LLANOSPARA OPTAR EL TITULO PROFESIONAL DE:
INGENIERO DE SISTEMAS

APROBADA POR EL JURADO REVISOR CONFORMADO POR:

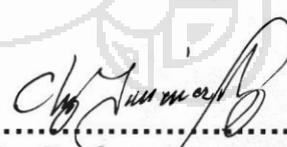
PRESIDENTE


Mg. Elmer Coylla Idme

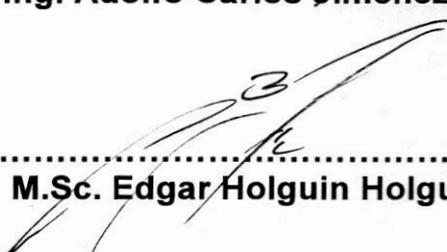
PRIMER MIEMBRO


M.Sc. Milder Zanabria Ortega

SEGUNDO MIEMBRO


Ing. Adolfo Carlos Jimenez Chura

DIRECTOR DE TESIS


M.Sc. Edgar Holguin HolguinÁREA: Informática
TEMA: comunicación de datosPuno – Perú
2014

DEDICATORIA

A mi mentora de vida y ejemplo a seguir,
mi madre: Dolores Llanos Pfura.



AGRADECIMIENTOS

Deseo expresar mis más sinceras muestras de agradecimientos a todas las personas que hicieron posible mi formación académica y personal.

Empezar por mi madre, por su amor, confianza, comprensión y apoyo durante toda mi formación personal y siempre estuvo a mi lado ante cualquier circunstancia. Jamás serán redundantes todas las muestras de mi eterno agradecimiento hacia ti.

A los docentes Mg. Elmer Coyla Idme, M.Sc. Milder Zanabria Ortega, e Ing. Adolfo Carlos Jiménez Chura, por sus consejos, observaciones, sugerencias y la guía hacia la finalización exitosa del presente trabajo.

Al M.Sc. Edgar Holguin Holguin, gracias a su desinteresado apoyo y empeño para el aporte del presente trabajo en la investigación científica. De igual modo a los ingenieros: Jaime Estela Gutiérrez, Avid Roman Gonzalez y Juan Martin Canales Romero, profesionales comprendidos a cerca de la Tecnología Aeroespacial, gracias por su asesoramiento y apoyo para la realización de la presente investigación, y motivación hacia la carrera espacial en nuestro país.

A todos mis docentes de la Escuela Profesional de Ingeniería de Sistemas, y a mis compañeros y amigos, no solo de escuela, sino de Facultad que somos, por haber compartido gratos momentos en lo académico y cultural.

Rudol Katari Huallpa Llanos

ÍNDICE

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN	10
INTRODUCCIÓN	11
CAPÍTULO I : PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACIÓN.....	14
1.1. PLANTEAMIENTO DEL PROBLEMA.....	15
1.2. FORMULACIÓN DEL PROBLEMA.....	16
1.3. ANTECEDENTES DE LA INVESTIGACIÓN.....	16
1.4. OBJETIVOS DE LA INVESTIGACIÓN.....	31
1.4.1. OBJETIVO GENERAL.....	31
1.4.2. OBJETIVOS ESPECÍFICOS.....	31
CAPÍTULO II : MARCO TEÓRICO, MARCO CONCEPTUAL E HIPÓTESIS DE LA INVESTIGACIÓN.....	32
2.1. MARCO TEÓRICO.....	33
2.1.1. SOFTWARE.....	33
2.1.2. SATELITES.....	45
2.1.3. CUBESAT.....	50
2.1.4. PROYECTO QB50.....	53



2.1.5.	ESTACIÓN TERRENA ESPACIAL.....	57
2.1.6.	TELEMETRÍA Y TELECOMANDO SATELITAL.....	59
2.2.	MARCO CONCEPTUAL.....	62
2.2.1.	CUBESAT.....	62
2.2.2.	DATO CRUDO.....	63
2.2.3.	ESTACIÓN TERRENA ESPACIAL.....	63
2.2.4.	MODELO EN CASCADA.....	63
2.2.5.	NANOSATÉLITE.....	64
2.2.6.	PROYECTO QB50.....	64
2.2.7.	SATÉLITES ARTIFICIALES.....	64
2.2.8.	SOFTWARE.....	64
2.2.9.	TELECONTROL.....	65
2.2.10.	TELEMETRÍA.....	65
2.3.	HIPÓTESIS DE LA INVESTIGACIÓN.....	65
2.3.1.	OPERACIONALIZACIÓN DE VARIABLES.....	66
CAPÍTULO III : MÉTODO DE INVESTIGACIÓN.....		67
3.1.	TIPO Y DISEÑO DE INVESTIGACIÓN.....	68
3.2.	POBLACIÓN Y MUESTRA.....	68
3.3.	MÉTODO DE RECOPIACIÓN DE DATOS.....	69
3.4.	MÉTODO DE PROCESAMIENTO DE DATOS.....	70
3.5.	MÉTODO DE ANÁLISIS DE DATOS.....	70
CAPÍTULO IV : RESULTADOS Y DISCUSIÓN.....		71

4.1. ESPECIFICACIÓN DEL SOFTWARE.....	72
4.1.1. DESCRIPCIÓN GENERAL.	72
4.1.2. REQUISITOS ESPECÍFICOS.	76
4.2. ANÁLISIS Y DISEÑO DEL PROCEDIMIENTO DE INTERPRETACIÓN DE DATOS CRUDOS Y LA PROGRAMACIÓN DE COMANDOS.....	85
4.2.1. PROCEDIMIENTO DE INTERPRETACIÓN DE DATOS CRUDOS.....	85
4.2.2. PROCEDIMIENTO PARA LA PROGRAMACIÓN DE COMANDOS.	90
4.3. DISEÑO E IMPLEMENTACIÓN DEL SOFTWARE.	91
4.3.1. DISEÑO.....	91
4.3.2. IMPLEMENTACIÓN.	111
4.4. EVALUACIÓN DEL SOFTWARE EN EL PROCESO DE TELEMETRÍA Y TELECONTROL.....	116
4.4.1. PRUEBA DE HIPÓTESIS.....	118
4.4.2. DECISIÓN.....	120
CONCLUSIONES	121
RECOMENDACIONES	123
BIBLIOGRAFÍA	125
ANEXOS	129

ÍNDICE DE FIGURAS

Figura N° 1. El Modelo en cascada. Fuente: Sommerville (2005).....	38
Figura N° 2. Arquitectura de flujo de datos. Fuente: Pressman, (2010).....	41
Figura N° 3. Estructura del nanosatélite Wara. Fuente: Verano, (2012).	50
Figura N° 4. Diagrama general de una estación terrena común. Fuente: Peredo, (2004).	59
Figura N° 5. Esquema de la estación terrena espacial – WARA. Fuente: elaboración propia.....	73
Figura N° 6. Datagrama o datos crudos recibidos del nanosatélite Wara. Fuente: elaboración propia.	86
Figura N° 7. Diagrama entidad – relación. Fuente: elaboración propia.....	93
Figura N° 8. Estructura lógica de la base de datos. Fuente: elaboración propia.....	94
Figura N° 9 Estructura física de la base de datos, archivo identificativos. Fuente: elaboración propia.	95
Figura N° 10 Estructura física de la base de datos, archivo nombre_satelite. Fuente: elaboración propia.	96
Figura N° 11 Diagrama de flujo de datos del software. Fuente: elaboración propia.....	98
Figura N° 12 Nivel 2 del diagrama de flujo de datos del software, para el proceso Autenticar datos de usuario. Fuente: elaboración propia.	99
Figura N° 13 Nivel 2 del diagrama de flujo de datos del software, para el proceso Interactuar con el usuario. Fuente: elaboración propia.....	101

Figura N° 14 Estructura del software para el proceso Interactuar con el usuario. Fuente: elaboración propia.....	102
Figura N° 15 Nivel 2 del diagrama de flujo de datos del software, para el proceso Procesar comandos. Fuente: elaboración propia.	103
Figura N° 16 Nivel 2 del diagrama de flujo de datos del software, para el proceso Administrar peticiones de mediciones. Fuente: elaboración propia.....	104
Figura N° 17 Estructura del software. Fuente: elaboración propia.	106
Figura N° 18 Icono del software. Fuente: elaboración propia.	107
Figura N° 19 Interfaz de acceso al software. Fuente: elaboración propia.	107
Figura N° 20 Interfaz de interacción con el usuario. Fuente: elaboración propia.....	108
Figura N° 21 Interfaz de selector de archivos. Fuente: elaboración propia.....	109
Figura N° 22 Interfaz de medición. Fuente: elaboración propia.	109
Figura N° 23 Interfaz de detalle de medición. Fuente: elaboración propia.....	110
Figura N° 24 Interfaz de controles. Fuente: elaboración propia.....	110
Figura N° 25 Instalador del software de telemetría y telecontrol. Fuente: elaboración propia.....	111
Figura N° 26 Pantalla de acceso. Fuente: elaboración propia.	112
Figura N° 27 Pantalla principal de menú. Fuente: elaboración propia.	112
Figura N° 28 Pantalla de medición. Fuente: elaboración propia.	113
Figura N° 29 Pantalla de medición en específico. Fuente: elaboración propia. ..	114
Figura N° 30 Pantalla de controles. Fuente: elaboración propia.	115
Figura N° 31. Grafico de matriz de datos. Fuente: elaboración propia.....	118

ÍNDICE DE TABLAS

Tabla N° 1. Clasificación de satélites según su masa. Fuente: García & Llorente (2013).....	47
Tabla N° 2. Operacionalización de variables. Fuente: elaboración Propia.....	66
Tabla N° 3. Muestra para la investigación. Fuente: elaboración propia.	69
Tabla N° 4. Características de los usuarios. Fuente: elaboración Propia.	74
Tabla N° 5. Interfaz de software: Java. Fuente: elaboración propia.....	76
Tabla N° 6. Interfaz de comunicación: protocolo TCP/IP. Fuente: Elaboración Propia.....	77
Tabla N° 7. Requisito funcional 1. Fuente: elaboración propia.	78
Tabla N° 8. Requisito funcional 2. Fuente: elaboración propia.	78
Tabla N° 9. Requisito funcional 3. Fuente: elaboración propia.	79
Tabla N° 10. Requisito funcional 4. Fuente: elaboración propia.	79
Tabla N° 11. Requisito funcional 5. Fuente: elaboración propia.	80
Tabla N° 12. Requisito funcional 6. Fuente: elaboración propia.	80
Tabla N° 13. Requisito funcional 7. Fuente: elaboración propia.	81
Tabla N° 14. Requisito funcional 8. Fuente: elaboración propia.	81
Tabla N° 15. Requisito funcional 9. Fuente: elaboración propia.	82
Tabla N° 16. Requisito funcional 10. Fuente: elaboración propia.	82
Tabla N° 17. Representación de caracteres en su valor decimal - 80. Fuente: elaboración propia.....	87
Tabla N° 18. Tabla de consistencia de resultados de pre-prueba. Fuente: elaboración propia.....	117

Tabla N° 19. Casos procesados. Fuente: elaboración propia. 117

Tabla N° 20. Estadístico de fiabilidad. Fuente: elaboración propia. 117

Tabla N° 21. Matriz de datos. Fuente: elaboración propia. 118

Tabla N° 22. Estadísticos de muestras relacionadas. Fuente: elaboración propia. 119

Tabla N° 23. Correlaciones de muestras relacionadas. Fuente: elaboración propia. 119

Tabla N° 24. Prueba de muestras relacionadas. Fuente: elaboración propia. 119



RESUMEN

El presente trabajo de investigación titulado “Software de Telecontrol y Telemetría para los procesos de Medición y Control del Nanosatélite Wara”, describe el desarrollo del software anunciado y la contribución de este en lo que será la Estación Terrena Espacial. Parte de los objetivos fue la especificación del software; el análisis y diseño del procedimiento para interpretar los datos crudos y así obtener los datos reales, lo mismo para realizar las órdenes y comandos, las cuales deben de empaquetarse en forma de dato crudo. Realizamos el diseño del software, con el diseño de datos y el diseño de la arquitectura del software; Se estableció que el software no tendrá un gestor de base de datos externo, utilizara archivos de texto plano (.dat) y la arquitectura del software fue diseñada mediante un modelo de flujo de datos. La implementación del software fue utilizando el lenguaje de programación JAVA, por ser multiplataforma, y el software es presentado en un paquete instalador. Se logra concluir y aceptar la Hipótesis planteada en el Proyecto de Investigación, gracias a la prueba de T de student. Y con el software aceptado, estamos listos para continuar con el desarrollo de nuestro primer satélite puneño.

Palabras Claves: Datos crudos, Estación Terrena Espacial, Software, Telemetría, Telecontrol o Telecomando.

INTRODUCCIÓN

El Nanosatélite Wara, nace a partir del proyecto Internacional QB50, el cual espera colocar en la órbita baja terrestre (LEO) una red de cincuenta nanosatélites, con el objetivo principal de explorar la termosfera baja, y la manera en que lograría desarrollar estos nanosatélites es con el apoyo de Universidades del mundo dispuestas a aprender y colaborar con crear sus propios satélites y afianzar sus conocimientos en la carrera espacial. De ahí, por el contacto de ingenieros peruanos científicos en estaciones espaciales de Europa, como son: Juan Martín Canales Romero, Jaime Estela Gutiérrez y Avid Román Gonzales, ofrecen la oportunidad a nuestra Universidad y así bajo resolución de Decanato N° 365-2011-D-FIMEES-UNA-P, se aprueba “LA EJECUCIÓN DEL PROYECTO QB-50 DE LA FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS”, para nuestra facultad es también conocido como el Proyecto WARA.

El desarrollo del nanosatélite Wara comprende el desarrollo de varios módulos, entre los principales el Módulo Central o computadora de abordaje, Módulo de Energía, Módulo de Control Térmico, Módulo de Comunicación, Módulo de Orientación, Estructura, Carga Útil o Experimento(s), Segmento de Tierra (centro de control y monitoreo, estación de antenas). Y a su vez, cada módulo tiene sub temas todos por investigar.

Actualmente ya se comenzó con el análisis y diseño del Módulo de Energía y el Módulo de orientación. Los cuales ya fueron sustentados en las Escuelas

profesionales de Ingeniería Mecánica Eléctrica e Ingeniería Electrónica, respectivamente.

El presente trabajo de investigación correspondería al Segmento de Tierra, o mejor dicho, a la Estación Terrena o centro de control y monitoreo, en dónde el desarrollo de aplicaciones y software se hace necesario para realizar estas tareas tan importantes, y específicamente para la medición y control del nanosatélite.

La investigación está debidamente estructurada, segmentado en capítulos donde desglosaremos con amplio detalle cada una de las partes compuestas.

Capítulo I. Comenzaremos por identificar el problema, explorando la necesidad de la existencia de una herramienta aplicativo para las funciones de Telemedición y Telecontrol, se estudió los antecedentes de la investigación, y nos planteamos el objetivo de la investigación, que es desarrollar un software para los procesos de Telemetría y Telecontrol, y evaluar su contribución.

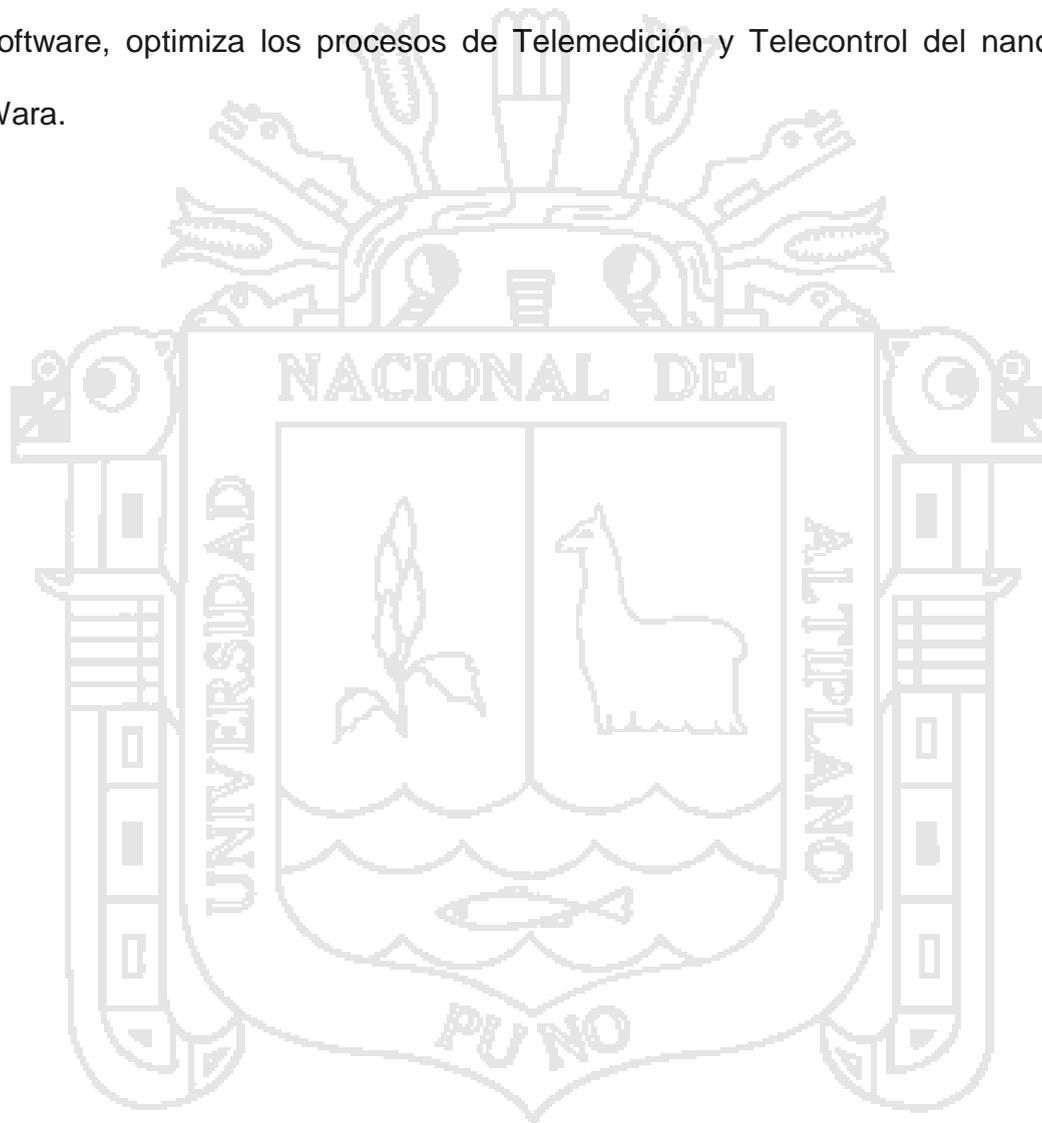
Capítulo II. Aquí efectuamos la revisión de la literatura, para nutrirnos de información teórica y conceptual para tener claro todos los conceptos necesarios.

Capítulo III. Concebimos el tipo y diseño de investigación, así como la metodología de investigación, los métodos de recolección, procesamiento de datos y análisis.

Capítulo IV. De resultados y discusión, donde se expone el desarrollo del software desde la especificación, diseño y su implementación, así también se analizó el procedimiento para interpretar los datos crudos y para la programación

de comandos. Para evaluar y validar el software se empleó la prueba estadística de correlación de muestras relacionadas.

Finalmente, el capítulo de Conclusiones y de Sugerencias nos detallaran las conclusiones obtenidas de la investigación, donde concluimos que el uso del software, optimiza los procesos de Telemedición y Telecontrol del nanosatélite Wara.



CAPÍTULO I : PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACIÓN



1.1. PLANTEAMIENTO DEL PROBLEMA.

La Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas (FIMEES) de la Universidad Nacional del Altiplano – Puno, viene impulsando un proyecto denominado Wara, el cual comprende la ejecución del Proyecto QB50 en la UNA Puno, concretamente consiste en el desarrollo de un nanosatélite de tipo 3U¹, el mismo que será lanzado aproximadamente en 2015, el nanosatélite actualmente está en proceso de desarrollo y construcción.

El modo de comunicación entre el nanosatélite y la Estación Terrena² será vía UHF (Ultra High Frequency). La información recolectada en el nanosatélite es enviada a tierra en forma de datos crudos. El tiempo de enlace de comunicación con el nanosatélite es un promedio de 10 minutos, tiempo suficiente para recibir toda la información que haya recabado el nanosatélite y reenviarle instrucciones al mismo.

Los datos crudos que se recibirán en la Estación Terrena Espacial, son poco entendibles y no muy fáciles de interpretar, aunque el operador de la estación terrena sepa interpretar estos valores, demandara tiempo en hacer esta tarea, incluso con riesgo a equivocarse en la interpretación haciendo una pésima medición de datos. De igual modo, pasara al momento de enviarle al nanosatélite instrucciones, para el proceso de telecontrol, tendría que estar digitando varios comandos corriendo el riesgo

¹ 3U, la estructura es de 3 unidades de Cubesat.

² Estación Terrena, es un ambiente con un conjunto de equipos y cómputo para establecer comunicación con satélites, se ampliara la información en los capítulos siguientes.

de errar en alguno de ellos desperdiciando tiempo en hacer el ingreso de comandos, ya que solo tenemos contacto con el nanosatélite un corto tiempo de minutos.

Teniendo en cuenta este problema, surge la necesidad de desarrollar una herramienta un apoyo para la estación terrena que controle y monitoree el nanosatélite, una manera más fácil y rápida de interpretación y medición de datos, que sea gráficamente entendible y así mismo, programar instrucciones de forma segura, sencilla y rápida para que en el tiempo de comunicación con el nanosatélite sea lo más provechosa posible, en conclusión hacer que el proceso de Telemetría y Telecontrol del nanosatélite sean óptimos.

1.2. FORMULACIÓN DEL PROBLEMA.

Luego de haber descrito el problema, se formuló la pregunta: Un software de análisis, interpretación y control, ¿Optimizara los procesos de Telemetría y Telecontrol del nanosatélite Wara?

1.3. ANTECEDENTES DE LA INVESTIGACIÓN.

Existen investigaciones hechas dentro de nuestra casa de estudio, parte del Proyecto Wara, así como de otras universidades, todas como aportes para el Proyecto QB50.

1.3.1. Verano Galindo, C. A., (2012), realizo la tesis: Diseño de un dinamo y Termo-Coupla como alternativas de generación de energía eléctrica para un nanosatélite tipo 3U Cubesat, en la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas de la Universidad Nacional del Altiplano Puno - Perú. La investigación llego a las conclusiones principales siguientes:

- El estudio referente al análisis de las fuentes de energía convencionales muestra un problema que valida el estudio para crear nuevas fuente de generación en el espacio, pues si consideramos que el Cubesat da unas 8 vueltas al día a nuestro planeta, eso quiere decir que de las 24 horas que tiene un día, 12 horas estará expuesto a la radiación solar y 12 horas estará pasando por la parte oscura de nuestro planeta.
- El módulo de generación magnético con un estimado de 4800 RPM, generaría algo más de 414.72 watts, lo cual es un valor importante si consideramos que el promedio de un Cubesat es solo de 6 watts, por otro lado el módulo de generación termoeléctrico, genera 1.49watts, si tenemos consideración de que el Cubesat puede llevar siete generadores termoeléctricos que generan cada uno este valor, vemos que este módulo puede generar 10.43 watts, valor que esta también por encima del consumo promedio del Cubesat.

Si bien es cierto esta investigación no tiene relación con el desarrollo de software, sino más bien es un referente de los componentes que tendrá el Nanosatélite, como es el caso de los generadores de energía eléctrica; Paneles Solares, el Dinamo y las Termo-couplas, son elementos los cuales producen datos, y dichos datos tienen que ser medidos e interpretados en la Estación Terrena.

1.3.2. Manzano Ramos E. A., (2013) realizó la Tesis: *Diseño e implementación del software para el cálculo de posición del picosatélite perteneciente al proyecto QB50 utilizando Mecánica Orbital*, en la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas de la Universidad Nacional del Altiplano Puno – Perú. La investigación llegó a las siguientes conclusiones principales:

- La Mecánica Orbital es una herramienta altamente eficaz y eficiente para el desarrollo del software que tenga como tarea el desarrollo de operaciones que van ligadas a ella, tales como cálculos de orbitas, posiciones, recorridos de satélites, naves, proyectiles, etc.
- El trabajo desarrollado por parte de un software o cualquier otra herramienta de cálculo digital para el caso de cálculo numérico implica un error siempre significativo por el caso de la coma flotante.
- EL algoritmo determinado para este proyecto no solamente implicó el uso de la Mecánica Orbital. Sino también de muchas

otras herramientas que forman parte del Marco Teórico y del Diseño de Ingeniería.

- El diseño de la IDE, implica el hecho de poder generar un semieje mayor que tenga un valor menor al Radio Terrestre Ecuatorial, para no generar orbitas que cortarían la superficie terrestre, lamentablemente este hecho no es tan sencillo, ya que excentricidades muy altas pueden generar orbitas, que a pesar de tener un semieje mayor alto corten a la superficie terrestre. Por lo tanto las operaciones de Cálculo de Coordenadas y Simulación determinarían casos con altitudes negativas y la operación de visibilidad afirmaría una no existente.

Esta investigación principalmente se centró en el diseño del algoritmo para el cálculo del posicionamiento geográfico, para poder determinar la Latitud, Longitud y Altitud de los satélites, en este caso para el proyecto WARA. Por lo que el software de Telecontrol y Telemetría no tendría que estudiar ya lo que respecta a la medición de la geo-posición del Nanosatélite.

1.3.3. Hernández Loja, M. A. & Rojas Advíncula, A. F., (2008) en su Trabajo de investigación: *Módulo De Comunicación Satelital*, perteneciente al Proyecto Chasqui-I en el Centro de Tecnologías de Información y Comunicaciones de la Universidad Nacional de Ingeniería, Lima – Perú. Obtuvieron las siguientes conclusiones:

- El ancho de banda más conveniente es el de radioaficionados.

- La modulación más usada para la realización de este tipo de comunicaciones espaciales es la FSK.
- El protocolo más usado y más eficiente para este tipo de enlaces es el AX.25.
- Los productos seleccionados tienen que ser probados intensamente para encontrar la mejor configuración ya sea el TNC, el modem, el transceiver, las antenas teniendo presente sus características eléctricas.
- Se ha podido determinar el tiempo aproximado en el cual se podrá entablar comunicación entre el nanosatélite y la estación terrena.
- Se ha podido determinar el alcance en la tierra para la realización de las descargar en otras palabras el ángulo de vista que se tendrá presente para realizar la comunicación con el nanosatélite.
- La potencia de transmisión no tiene que ser muy alta, para que se dé una buena comunicación.

Este trabajo se basa la viabilidad de la comunicación del Satélite con la Estación Terrena, y está separada en dos tareas principales: el Diseño de Comunicación, y la Telemetría, Seguimiento y Comando.

Este último es el que se tomó en cuenta, ya que menciona que estas funciones son vitales para el éxito de la operación del nanosatélite, las principales funciones serán las siguientes: Supervisar la ejecución de todos

los subsistemas del nanosatélite, esto recibiendo los datos de seguimiento en el Centro de Control de la estación terrena. Y enviar comandos del Centro de Control de la estación terrena para la ejecución de diversas funciones del satélite.

1.3.4. Roman Gonzales, A. & Vargas Cuentas, N. I. (2013), en su artículo científico: *Tecnología Aeroespacial en el Mundo*, Comentan en su resumen que el desarrollo aeroespacial para un país representa su nivel tecnológico, su capacidad de desarrollo y en cierta manera su poderío. El desarrollo de la industria aeroespacial trae consigo mucha tecnología para el uso del gran público en nuestros días cotidianos, tecnologías desarrolladas inicialmente para explorar el espacio y que hoy en día son utilizadas en el diario vivir. Las conclusiones y recomendaciones de este artículo son:

- Se puede observar que el desarrollo de la tecnología aeroespacial no solo sirve para explorar el espacio, si no que paralelo a ello nos trae muchas aplicaciones para nuestro diario vivir en la Tierra.
- La posición geográfica del Perú, debido a su cercanía a la Línea Ecuatorial, es algo que se debe aprovechar para fomentar el desarrollo de la actividad de lanzamientos y poder atraer a las potencias en el rubro para firmar convenios que sean beneficiosos ampliamente.

- Las imágenes satelitales son de gran ayuda en muchos sectores, no solo para la exploración espacial, sino también para conocer nuestro planeta, para ayudar a nuestras actividades diarias y ayuda en los sectores que mueven la economía del Perú como son la agricultura y la minería.
- La implementación de nanosatélites es una buena manera de empezar en el área de lo aeroespacial, es bueno que los chicos universitarios puedan participar en proyectos de ese tipo. También se recomendaría que se creen en las universidades la Carrera Profesional de Ingeniería Aeroespacial.
- De manera general, el desarrollo de tecnología aeroespacial es de mucha ayuda y trae mucho desarrollo al país que decide invertir en dicho campo, puede servir como herramienta de inclusión social, herramienta para control militar, manejo de zonas mineras, manejo de la contaminación, etc. Por ello se recomendaría que los gobiernos puedan decidir el apoyo al desarrollo de este tipo de tecnología.

Es un trabajo que principalmente trata de la importancia que tiene la tecnología aeroespacial y su implicancia, podríamos mencionar las telecomunicaciones satelitales, los sistemas de posicionamiento y navegación, telemedicina, y otros que son de utilidad en nuestra vida cotidiana.

Nos informa que el Perú cuenta con una agencia espacial, la Comisión Nacional de Investigación y Desarrollo Aeroespacial (CONIDA), también contamos con un Centro de Lanzamientos de Punta Lobos. El Perú tiene una herencia reconocida a nivel mundial que se extiende desde épocas pre-incaicas hasta nuestros días. Descubrimientos astronómicos sobre los que se sustentó la agricultura; las avanzadas técnicas quirúrgicas y las obras monumentales de ingeniería desarrolladas durante la época pre-hispánica; participación histórica de destacados peruanos en los primeros años del desarrollo de la aeronáutica. El Ing. Pedro Paulet Mostajo es considerado como uno de los pioneros en lo que a cohetes se refiere, por su motor a propulsión de combustible líquido.

Y en buena manera motiva a los estudiantes universitarios a iniciarse en el campo aeroespacial, comenzando con el desarrollo de nanosatélites, teniendo en cuenta siempre que los objetivos de esto son netamente académicos.

1.3.5. Hillán García, L., Setién Doderó, F. y Del Real Colomo, A. (2014) autores del artículo científico: *El Sistema de Telemedicina Militar en España: una aproximación histórica*, publicado en la revista Sanidad Militar en Madrid edición: abril – Junio de 2014. Describen las siguientes conclusiones:

- Nadie parece dudar del potencial de la telemedicina, y de las oportunidades que ésta puede ofrecer en la provisión de la cobertura sanitaria, la equidad en el acceso, la calidad en los

servicios o la eficiencia en la utilización de los recursos médicos.

A la luz de este potencial, la Organización Nacional de la Salud (OMS) abrió el Observatorio Global para la eHealth (GOe) con el objetivo de analizar los beneficios que las TIC pueden traer a la asistencia médica y al bienestar del paciente.

- Sin embargo, aún quedan retos y barreras que superar entorno al uso de la telemedicina, entre los que se suelen citar: la resistencia a la adopción de modelos de servicios innovadores; la carencia de formación en aplicaciones de telemedicina; la falta de estudios que demuestren los beneficios económicos y la rentabilidad de la telemedicina, y la ausencia de un marco jurídico internacional que permita a los profesionales de la salud prestar servicios en diferentes jurisdicciones y países. Y a pesar de estas limitaciones, la telemedicina ha experimentado, en un corto periodo de tiempo, un rápido crecimiento acompañado por los avances en las TIC e intensificado por unas necesidades y demanda del paciente cada vez mayores a los que dar respuesta.
- Un buen ejemplo del importante papel que ejerce la telemedicina en la sanidad lo encontramos en las Fuerzas Armadas Españolas; el gran desarrollo que ha experimentado esta disciplina en el mundo militar en poco años, nos da una idea de las enormes ventajas que puede incorporar la telemedicina a la sanidad, y el prometedor futuro que le queda por delante. Las

comunicaciones vía satélite con centros remotos de difícil acceso, supuso un gran salto en la Sanidad Militar, y posicionó a su Sistema de Telemedicina como el mejor catalizador en la provisión de una atención sanitaria sostenible y de calidad en cualquier momento y en cualquier parte del planeta.

- En la actualidad, se están realizando grandes inversiones en telemedicina, como demuestra su creciente importancia económica, dentro y fuera del ámbito militar. Sin embargo hay que tener en cuenta que el grado de desarrollo, y aplicación de la telemedicina difiere en función de las necesidades y funcionamiento de quién la provee y utiliza, así por ejemplo, en España, las Fuerzas Armadas españolas se sitúan al frente de esta práctica con su Sistema de Telemedicina que da respuesta y cubre las exigencias sanitarias en el ejercicio diario de sus actividades. No en vano, este Sistema de Telemedicina es considerado uno de los pilares en los que se sustenta la Sanidad Militar, y un referente internacional en telemedicina miliar.
- Por tanto, no parece difícil imaginar hacia dónde se fijarán y orientarán las nuevas agendas políticas, cuando ante todas las miradas, la telemedicina se presenta como una solución atractiva y factible para la sanidad de todas las naciones. Son pocas las cuestiones en las que existe un acuerdo unánime e internacional, máxime en las condiciones adversas en las que se encuentra la economía mundial, es por esto, que cabe pensar

que asistiremos a un desarrollo de la telemedicina aún más impactante que el acontecido hasta el momento, sobre todo si como es de esperar, la telemedicina adopta y aprovecha los continuos avances en las tecnologías de la información y las comunicaciones.

Este artículo científico destaca como el empleo de las TIC's ha ofrecido muchas ventajas para la Telemedicina, narra cómo es que Telemedicina rompe las barreras tiempo-espacio para poder ofrecer asistencia sanitaria cuándo y dónde se necesite, y que con los avances de las TIC, la optimización y mejora continua es cada vez más significativa, abriéndose atractivos horizontes y grandes expectativas. Sobre todo informa que la manera de realizar estas asistencias médicas en cualquier parte del mundo, es estableciendo sesión de videoconferencia vía Satélite, un motivo más para comenzar con el desarrollo de la carrera Aeroespacial.

1.3.6. Peccia, N. M., autor del paper: *A brief Story of a success: The CCSDS*, publicado por The American Institute of Aeronautics and Astronautics, Inc., durante la SpaceOpsConferences durante el 5 y 9 de Mayo de 2014 en Pasadena, CA. Donde tiene su única conclusión:

- En este trabajo se ha presentado el Plan Estratégico del CCSDS actual, dando una visión del conjunto de sus áreas técnicas, las Normas y Prácticas Recomendadas y las más innovadoras Normas de desarrollo.

El CCSDS (de las siglas en inglés: The Consultative Committee for Space Data Systems), es el Comité Consultivo de Sistemas Espaciales de Datos, que da a conocer su Plan Estratégico. El CCSDS es una organización internacional apoyada por 39 agencias espaciales y cerca de 145 empresas, se formó en 1982 con el fin de proporcionar un foro para la solución de los problemas comunes en el desarrollo y operaciones de sistemas espaciales de datos. El CCSDS está destinado a la innovación durante la evolución de la tecnología y el proceso de desarrollo, la validación, el mantenimiento y la promoción de un cuerpo único de normas de sistemas de datos espaciales, que se centra en la interoperabilidad de los sistemas espaciales y de apoyo entre las organizaciones espaciales.

Resaltamos de este artículo la especificación de objetivos de cada Área, entre ellos: el Área de Ingeniería de Sistemas, el área del servicio de apoyo de la interfaz a bordo, el área de servicio de enlace en el espacio, entre otros, y el área de servicio de operaciones de la misión y gestión de la información. Este último por ejemplo, uno de sus objetivos es Especificar ampliamente un conjunto de normas recomendadas para las Operaciones de la Misión (MO), Servicios que se exponen para la interoperabilidad y soporte o usados internamente por una agencia y que cubra al menos un 70% de las principales interfaces. Para ello existen normas recomendadas: usar el XML Telemetric and Command Exchange (XTCE), en trabajos futuro las normas están previstas para la automatización, la gestión de archivos, la navegación, la planificación, la administración de Buffer, la

programación, gestión de Software y servicios de tiempo. Resulto muy interesante la estandarización de los datos, con el uso del XML.

1.3.7. Houlihan, S. & Kalman, A. (2014). Autores del Paper: *Applying Standard Commands for Programmable Instruments (SCPI) to CubeSats*, publicado en: Pumpkin Inc. San Francisco. Obtuvo las siguientes conclusiones:

- Aplicando los estándares SCPI para módulos CubeSat proporcionan un método robusto para el comando y telemetría, que es muy fácil de aprender para operadores y programadores mientras es lo suficientemente potente para el uso en operaciones de órbita. Una cuidadosa atención a la arquitectura de hardware y software asegura que haya una máxima interoperabilidad entre módulos y una cantidad mínima de esfuerzo requerido para añadir el soporte para los nuevos módulos de software de vuelo.
- El uso del bus I²C es el método más sencillo para conectar eléctricamente un sistema multi-nodo.
- Un mayor desarrollo de la familia de MCU Supervisor de módulos CubeSat añade la funcionalidad existente.
- Se espera que la cantidad de diferentes tipos de módulos Supervisor MCU disponibles, crezca más allá de la GPSR y SIM que han sido diseñados. La aprobación de SCPI comandos comunes estándares de interfaz de módulos CubeSat, deben

reducir de forma significativa tiempo necesario para las pruebas, la integración y el desarrollo de software para las futuras misiones CubeSat.

Se presenta en este paper la aplicación de Comandos Estándar para Instrumentos Programables (siglas en inglés SCPI) que se aplican como Supervisor de módulos para los nanosatélites. SCPI Incluye humanos de legibilidad y reporte de errores estándar, lo que puede ser beneficioso en una variedad de circunstancias, incluyendo prototipos pruebas y supervisión de tiempo de ejecución. También se presenta una visión general del software y el hardware necesario para el soporte con SCPI, y su rendimiento en tiempo de ejecución en una producción CubeSat.

1.3.8. Riot, V., Simms, L., Carter, D., Decker, T., Newman, J., Magallanes, L., ... & Williamson, D. (2014). Son autores del paper: *Government-owned CubeSat Next Generation Bus Reference Architecture*, publicado durante la 28th Annual AIAA/USU Conference on Small Satellites. Aquí se llegó a las siguientes conclusiones:

- La arquitectura CNGB se basa en los últimos programas de bus CubeSat y proporciona la flexibilidad y transparencia necesarias para la comunidad, así como aprobación del sector.
- Este programa tiene por objetivo proporcionar el marco necesario para desarrollar una base de intercambios de productos comerciales que se pueden seleccionar para confeccionar un CubeSat o en el diseño de satélites pequeños

para una aplicación específica con una disminución de la necesidad de desarrollo personalizado.

- Este programa también proporciona la fundación para una base de software que puede crecer y enriquecerse como los programas que contribuyen a las capacidades disponibles de implementación.
- Los equipos de desarrollo CubeSat pueden optar por cumplir con la arquitectura de software, arquitectura eléctrica, arquitectura mecánica o cualquier combinación basada en sus necesidades.
- La arquitectura CNGB también proporciona flexibles herramientas de prueba para el desarrollo ágil, integración y pruebas.

Este último paper comenta que la Lawrence Livermore National Laboratory (LLNL), en colaboración con la Naval Postgraduate School (NPS) está desarrollando un CubeSat bus referido a la arquitectura y un conjunto de especificaciones mínimas útiles para aplicaciones gubernamentales. La arquitectura tiene aplicaciones de software, eléctricas y mecánicas. Las interfaces y tiene como objetivo proporcionar una plataforma flexible que puede ser aprobada por la industria, apoyando el intercambio de los componentes mientras mantiene la personalización de integración de cargas útiles.

En el punto Interfaz de Software, se espera que la arquitectura de software, deba soportar en intercambio de hardware, es decir, ser

compatible con una amplia gama de plataformas informáticas (desde pequeños micro controladores a los procesadores de varios núcleos procesadores grandes) y que permiten un modelo de código abierto. Además, se espera que los recursos informáticos en un CubeSat o un satélite pequeño sean distribuidos a través de diferentes ubicaciones físicas. Se describe también que cada módulo que se implemente de un dispositivo de hardware debe proporcionar un xTEDS, que no es otra cosa que estructurarlo mediante XML.

1.4. OBJETIVOS DE LA INVESTIGACIÓN.

Los objetivos de la investigación fueron.

1.4.1. OBJETIVO GENERAL.

Realizar un Software para interpretar y analizar los datos crudos, y para la programación de comandos, para los procesos de Telemetría y Telecontrol del Nanosatélite Wara, y evaluar su desempeño.

1.4.2. OBJETIVOS ESPECÍFICOS.

- Realizar la especificación del Software.
- Analizar y diseñar el procedimiento de interpretación de datos crudos y la programación de comandos.
- Diseñar e implementar el Software.
- Evaluar el software en el proceso de Telecontrol y Telemetría del nanosatélite Wara para probar la hipótesis.

CAPÍTULO II : MARCO TEÓRICO, MARCO CONCEPTUAL E HIPÓTESIS DE LA INVESTIGACIÓN



2.1. MARCO TEÓRICO.

2.1.1. SOFTWARE.

Según Sommerville(2005, Pag. 5). “el software no son sólo programas, sino todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta”.

El software se puede definir como el conjunto de tres componentes: *Programas* (instrucciones), este componente proporciona la funcionalidad deseada y el rendimiento cuando se ejecute; *Datos*, este componente incluye los datos necesarios para manejar y probar los programas y las estructuras requeridas para mantener y manipular estos datos, y *Documentos*, este componente describe la operación y uso del programa (Instituto Nacional de Tecnologías de la Comunicación, 2009, pág. 8).

Por lo general, un sistema de software consiste en diversos programas independientes, archivos de configuración que se utilizan para ejecutar estos programas, un sistema de documentación que describe la estructura del sistema, la documentación para el usuario que explica cómo utilizar el sistema y sitios web que permitan a los usuarios descargar la información de productos recientes. Existen dos tipos de productos de software:

- *Productos genéricos.* Son sistemas aislados producidos por una organización de desarrollo y que se venden al mercado abierto a cualquier cliente que le sea posible comprarlos. Ejemplos de este tipo de producto son el software para computadoras tales como bases de datos, procesadores de texto, paquetes de dibujo y herramientas de gestión de proyectos.
- *Productos personalizados (o hechos a medida).* Son sistemas requeridos por un cliente en particular. Un contratista de software desarrolla el software especialmente para ese cliente. Ejemplos de este tipo de software son los sistemas de control para instrumentos electrónicos, sistemas desarrollados para llevar a cabo procesos de negocios específicos y sistemas de control del tráfico aéreo.

Una diferencia importante entre estos diferentes tipos de software es que, en los productos genéricos, la organización que desarrolla el software controla su especificación. La especificación de los productos personalizados, por lo general, es desarrollada y controlada por la organización que compra el software. Los desarrolladores de software deben trabajar con esa especificación.

PROCESO DEL SOFTWARE.

Un proceso del software es un conjunto de actividades y resultados asociados que producen un producto de software (Sommerville, 2005, pág. 60). Estas actividades pueden consistir en

el desarrollo de software desde cero en un lenguaje de programación estándar como Java o C. Sin embargo, cada vez más, se desarrolla nuevo software ampliando y modificando los sistemas existentes y configurando e integrando software comercial o componentes del sistema.

Los procesos del software son complejos y, como todos los procesos intelectuales y creativos, dependen de las personas que toman decisiones y juicios. Debido a la necesidad de juzgar y crear, los intentos para automatizar estos procesos han tenido un éxito limitado. Las herramientas de ingeniería del software asistida por computadora (CASE) pueden ayudar a algunas actividades del proceso. Sin embargo, no existe posibilidad alguna, al menos en los próximos años, de una automatización mayor en el diseño creativo del software realizado por los ingenieros relacionados con el proceso del software.

Una razón por la cual la eficacia de las herramientas CASE está limitada se halla en la inmensa diversidad de procesos del software. No existe un proceso ideal, y muchas organizaciones han desarrollado su propio enfoque para el desarrollo del software. Los procesos han evolucionado para explotar las capacidades de las personas de una organización, así como las características específicas de los sistemas que se están desarrollando. Para algunos sistemas, como los sistemas críticos, se requiere un proceso de desarrollo muy estructurado. Para

sistemas de negocio, con requerimientos rápidamente cambiantes, un proceso flexible y ágil probablemente sea más efectivo.

Aunque existen muchos procesos diferentes de software, algunas actividades fundamentales son comunes para todos ellos:

- 1) *Especificación del software.* Se debe definir la funcionalidad del software y las restricciones en su operación.
- 2) *Diseño e implementación del software.* Se debe producir software que cumpla su especificación.
- 3) *Validación del software.* Se debe validar el software para asegurar que hace lo que el cliente desea.
- 4) *Evolución del software.* El software debe evolucionar para cubrir las necesidades cambiantes del cliente.

Diferentes tipos de sistemas necesitan diferentes procesos de desarrollo. Por lo tanto, estas actividades genéricas pueden organizarse de diferentes formas y describirse en diferentes niveles de detalle para diferentes tipos de software. Sin embargo, el uso de un proceso inadecuado del software puede reducir la calidad o la utilidad del producto de software que se va a desarrollar y/o incrementar los costes de desarrollo.

MODELOS DEL PROCESO DEL SOFTWARE.

Un modelo del proceso del software es una representación abstracta de un proceso del software. Cada modelo de proceso

representa un proceso desde una perspectiva particular, y así proporciona sólo información parcial sobre ese proceso. Existen varios modelos de proceso muy generales y se presentan desde una perspectiva arquitectónica. Esto es, vemos el marco de trabajo del proceso, pero no los detalles de actividades específicas.

Estos modelos generales no son descripciones definitivas de los procesos del software. Más bien, son abstracciones de los procesos que se pueden utilizar para explicar diferentes enfoques para el desarrollo de software. Puede pensarse en ellos como marcos de trabajo del proceso que pueden ser extendidos y adaptados para crear procesos más específicos de ingeniería del software.

Los modelos de procesos generales son:

El Modelo en Cascada

O también llamado Modelo Lineal Secuencial, el primer modelo de proceso de desarrollo de software que se publicó se derivó de procesos de ingeniería de sistemas más generales. Este modelo se muestra en la **¡Error! No se encuentra el origen de la referencia.** debido a la cascada de una fase a otra, dicho modelo se conoce como modelo en cascada o como ciclo de vida del software

(Sommerville, 2005, pág. 62). Las principales etapas de este modelo se transforman en actividades fundamentales de desarrollo:

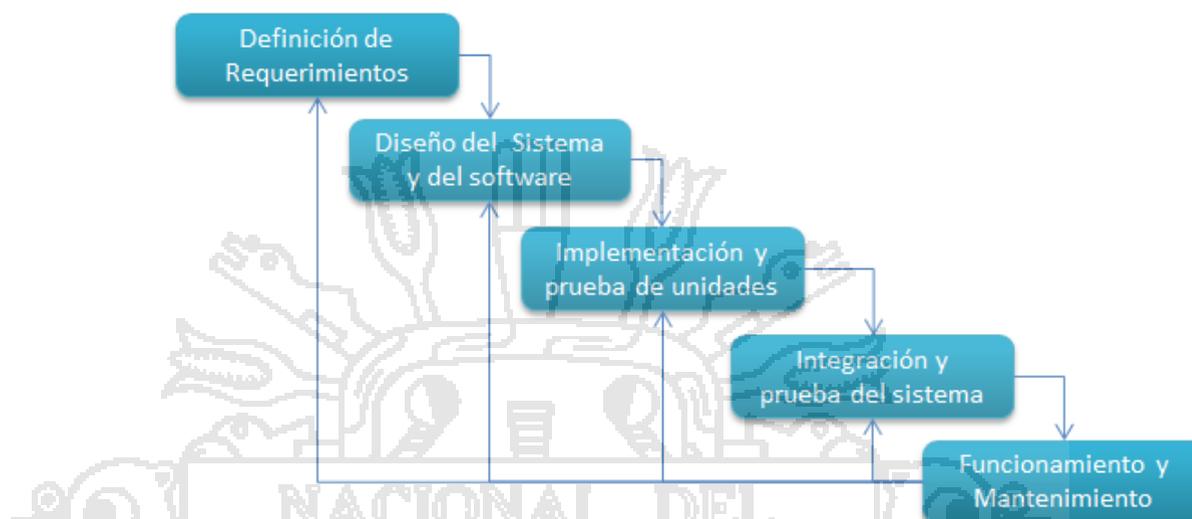


Figura N° 1. El Modelo en cascada. Fuente: Sommerville (2005).

- 1) **Análisis** y definición de requerimientos. Los servicios, restricciones y metas del sistema se definen a partir de las consultas con los usuarios. Entonces, se definen en detalle y sirven como una especificación del sistema.
- 2) **Diseño del sistema y del software**. El proceso de diseño del sistema divide los requerimientos en sistemas hardware o software. Establece una arquitectura completa del sistema. El diseño del software identifica y describe las abstracciones fundamentales del sistema software y sus relaciones.
- 3) **Implementación y prueba de unidades**. Durante esta etapa, el diseño del software se lleva a cabo como un conjunto o unidades de programas. La prueba de unidades implica verificar que cada una cumpla su especificación.

4) **Integración y prueba del sistema.** Los programas o las unidades individuales de programas se integran y prueban como un sistema completo para asegurar que se cumplan los requerimientos del software. Después de las pruebas, el sistema software se entrega al cliente.

5) **Funcionamiento y mantenimiento.** Por lo general, ésta es la fase más larga del ciclo de vida. El sistema se instala y se pone en funcionamiento práctico. El mantenimiento implica corregir errores no descubiertos en las etapas anteriores del ciclo de vida, mejorar la implementación de las unidades del sistema y resaltar los servicios del sistema una vez que se descubren nuevos requerimientos.

En principio, el resultado de cada fase es uno o más documentos aprobados. La siguiente fase no debe empezar hasta que la fase previa haya finalizado. En la práctica, estas etapas se superponen y proporcionan información a las otras.

Las ventajas del modelo en cascada son que la documentación se produce en cada fase y que éste cuadra con otros modelos del proceso de ingeniería. Su principal problema es su inflexibilidad al dividir el proyecto en distintas etapas. Se deben hacer compromisos en las etapas iniciales, lo que hace difícil responder a los cambios en los requerimientos del cliente.

Durante el diseño se identifican 4 elementos principales: diseño de datos, arquitectura, componentes e interfaz. El diseño de la arquitectura del software es el equivalente del plano de una casa. Éste ilustra la distribución general de las habitaciones, su tamaño, forma y relaciones entre ellas, así como las puertas y ventanas que permiten el movimiento entre los cuartos. El plano da una visión general de la casa. Los elementos del diseño de la arquitectura dan la visión general del software. (Pressman, 2010).

El modelo arquitectónico proviene de tres fuentes: 1) información sobre el dominio de la aplicación del software que se va a elaborar, 2) los elementos específicos del modelo de requerimientos, tales como diagramas de flujo de datos o clases de análisis, sus relaciones y colaboraciones para el problema en cuestión, y 3) la disponibilidad de estilos arquitectónicos y sus patrones.

Pressman (2010), nos señala que entre los estilos y patrones arquitectónicos comunes para el software existen:

- **Arquitectura centrada en los datos.** En el centro de esta arquitectura se halla un almacenamiento de datos (como un archivo o base de datos) al que acceden con frecuencia otros componentes que actualizan, agregan, eliminan o modifican los datos de cierto modo dentro del almacenamiento.

- **Arquitecturas de flujo de datos.** Esta arquitectura se aplica cuando datos de entrada van a transformarse en datos de salida a través de una serie de componentes computacionales o manipuladores. Tiene un conjunto de componentes, llamados filtros, conectados por tubos que transmiten datos de un componente al siguiente. Se diseña para esperar una entrada de datos de cierta forma y produce datos de salida (al filtro siguiente) en una forma especificada. Sin embargo, el filtro no requiere ningún conocimiento de los trabajos que realizan los filtros vecinos, es decir trabaja de forma independiente.

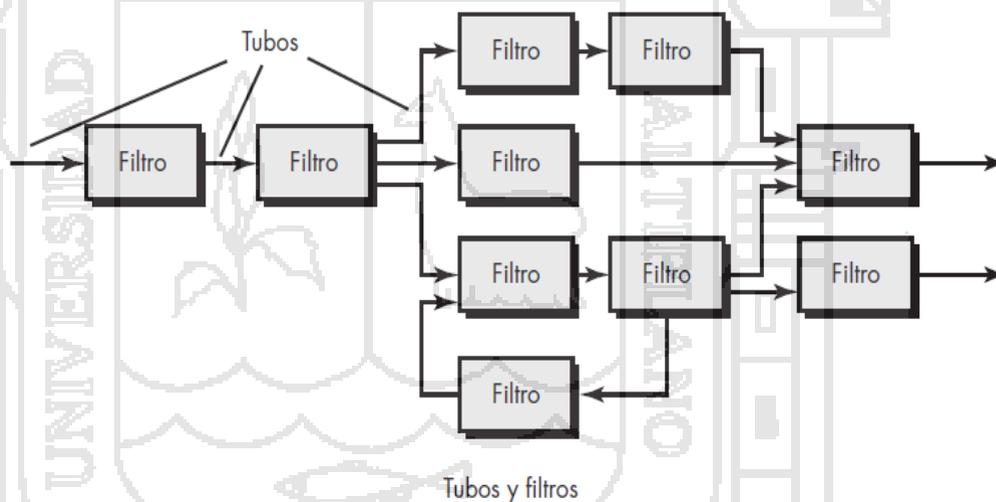


Figura N° 2. Arquitectura de flujo de datos. Fuente: Pressman, (2010).

- **Arquitecturas de llamar y regresar.** Dentro de esta categoría están las: Arquitecturas de programa principal/subprograma, esta estructura descompone una función en una jerarquía de control en la que un programa “principal” invoca cierto número de componentes de programa que a su vez invocan a otros. Y la Arquitecturas de llamada de procedimiento remoto, los

componentes de una arquitectura de programa principal/subprograma están distribuidos a través de computadoras múltiples en una red.

- **Arquitecturas orientadas a objetos.** Los componentes de un sistema incluyen datos y las operaciones que deben aplicarse para manipularlos. La comunicación y coordinación entre los componentes se consigue mediante la transmisión de mensajes.
- **Arquitecturas en capas.** Se define un número de capas diferentes; cada una ejecuta operaciones que se aproximan progresivamente al conjunto de instrucciones de máquina. En la capa externa, los componentes atienden las operaciones de la interfaz de usuario. En la interna, los componentes realizan la interfaz con el sistema operativo. Las capas intermedias proveen servicios de utilerías y funciones de software de aplicación.

INGENIERÍA DEL SOFTWARE ASISTIDA POR COMPUTADORA.

Ingeniería del Software Asistida por Computadora (CASE) es el nombre que se le da al software que se utiliza para ayudar a las actividades del proceso del software como la ingeniería de requerimientos, el diseño, el desarrollo de programas y las pruebas. Por lo tanto, las herramientas CASE incluyen editores de diseño, diccionarios de datos, compiladores, depuradores, herramientas de construcción de sistemas, etcétera.

La tecnología CASE está disponible para la mayoría de las actividades rutinarias en el proceso del software. Esto permite algunas mejoras en la calidad y productividad del software. Las mejoras por la utilización de CASE están limitadas por dos factores: Esencialmente, la ingeniería del software es una actividad de diseño que se basa en la creatividad. Los sistemas CASE automatizan las actividades rutinarias, pero los intentos de utilizar la inteligencia artificial para proporcionar ayuda al diseño no han tenido éxito. En la mayoría de las organizaciones, la ingeniería del software es una actividad de equipo, y los ingenieros invierten mucho tiempo interactuando con los otros miembros del equipo. La tecnología CASE no proporciona mucha ayuda para esto. Actualmente, la tecnología CASE está madura y hay herramientas disponibles y bancos de trabajo de un amplio rango de proveedores.

Existen varias formas diferentes de clasificar las herramientas CASE, cada una de las cuales nos proporciona una perspectiva distinta de estas herramientas. Una clasificación de las herramientas CASE acorde con su función, se muestra a continuación, donde se enumera diferentes tipos de herramientas CASE. Ésta no es una lista completa de herramientas CASE, las herramientas especializadas, como las de ayuda a la reutilización, no se incluyen.

- *Herramientas de planificación*, de estimación, PERT, hojas de cálculo.
- *Herramientas de edición*, editores de texto, de diagramas, procesadores de texto.
- *Herramientas de gestión del cambio*, de rastreo de requerimientos.
- *Herramientas de gestión de la configuración*, sistemas de gestión de las versiones, herramientas de construcción de sistemas.
- *Herramientas de construcción de prototipos*, lenguajes de muy alto nivel, generadores de interfaz de usuario.
- *Herramientas de apoyo a métodos*, editores de diseño, generadores de código, diccionarios de datos.
- *Herramientas de procesamiento de lenguajes*, compiladores e intérpretes.
- *Herramientas de análisis de programas*, generadores de referencias cruzadas, analizadores estáticos y dinámicos.
- *Herramientas de pruebas*, generadores de pruebas de datos, comparadores de archivos.
- *Herramientas de depuración*, sistemas de depuración interactiva.
- *Herramientas de documentación*, Programas de diseño de páginas, editores de imágenes.
- *Herramientas de reingeniería*, sistemas de referencias cruzadas, sistemas reestructuración de programas.

2.1.2. SATELITES.

La National Aeronautics and Space Administration [NASA], (2011), define que: “Un satélite es un objeto que se mueve alrededor de un objeto más grande”.

Explícitamente es cualquier objeto que gravita en una órbita cerrada alrededor de un planeta, generalmente el satélite es mucho más pequeño y acompaña al planeta en su traslación alrededor de la estrella que orbita.

SATÉLITES NATURALES.

Los satélites naturales son objetos que como su nombre indica fueron creados por la naturaleza y no intervino la mano del hombre. Un claro ejemplo es la Luna, podría considerarse como un sistema de dos planetas que orbitan juntos (sistema binario de planetas). Similar es el caso de Plutón y su satélite Caronte. Si dos objetos poseen masas similares, se suele hablar de sistema binario en lugar de un objeto primario y un satélite. El criterio habitual para considerar un objeto como satélite es que el centro de masas del sistema formado por los dos objetos esté dentro del objeto primario. El punto más elevado de la órbita del satélite se conoce como apoápside.

SATÉLITES ARTIFICIALES.

Los satélites artificiales son máquinas hechas por el hombre. Estas máquinas son lanzadas al espacio y orbitan la Tierra u otro cuerpo en el espacio. Hay miles de satélites artificiales, algunos toman fotografías de nuestro planeta, algunos toman fotografías de otros planetas, el sol y otros objetos. Estas imágenes ayudan a los científicos a aprender sobre la Tierra, el sistema solar y el universo (NASA, 2011).

El primer satélite artificial fue el Sputnik I lanzado por la Unión Soviética el 4 de octubre de 1957. Desde entonces se han colocado en órbita miles de satélites artificiales muchos de los cuales aún continúan en órbita alrededor de la Tierra con gran variedad de fines, científicos, tecnológicos y militares.

En la actualidad los satélites artificiales juegan un papel sumamente importante en nuestra civilización, otorgándonos desde la capacidad de comunicarnos de manera casi instantánea en cualquier parte del globo terrestre (telecomunicaciones satelitales), brindándonos nuestra posición exacta en cualquier parte del planeta (Sistema de Posicionamiento Global GPS) e inclusive permitiéndonos predecir hasta cierto punto las condiciones climáticas alrededor de todo el mundo.

Dependiendo de su aplicación los satélites varían en tamaño y masa. Los hay desde aquellos de varios metros de envergadura, como por ejemplo el Telescopio Espacial Hubble, hasta los de pocos centímetros, como los empleados en el proyecto *Cubesat*. De acuerdo a su masa los satélites se clasifican de la siguiente manera. (García & Llorente, 2013):

Clasificación	Masa Neta
Satélite Grande	>1,000Kg
Satélite Mediano	500-1,000Kg
Minisatélite	100-500Kg
Microsatélite	10-100Kg
Nanosatélite	1-10Kg
Picosatélite	0.1-1Kg
Femtosatélite	<0.1Kg

Tabla N° 1. Clasificación de satélites según su masa. Fuente: García & Llorente (2013).

NANOSATÉLITE.

Como se describió anteriormente, es una clasificación de satélite según su masa. Los satélites artificiales en general cuentan con las mismas partes principales, ya sean para misiones de exploración temporales, o para misiones de larga duración. A continuación se describe en detalle las partes más importantes de un Nanosatélite (Mozombite Frisancho, 2012, pág. 13).

- **Estructura.** La estructura mecánica es la que le da forma al satélite, se encarga de albergar todas las unidades funcionales del mismo, así como de protegerlas del hostil ambiente del

vacío. Para el diseño de la estructura mecánica se prefiere materiales que tengan gran resistencia y a la vez que tengan poca masa, ya que contemplando las limitaciones de la misma, una estructura demasiado pesada podría hacer necesario remover ciertas partes electrónicas al sistema. A la vez de ser el armazón en donde toda la electrónica y los experimentos del satélite se resguardan, comúnmente se utilizan las caras de la estructura para colocar las celdas fotovoltaicas, en reemplazo de los sistemas de celdas solares desplegadas, que se observan comúnmente en satélites de mayor tamaño, esto se lleva a cabo en general porque los Nanosatélites no tienen un gran consumo de potencia, por lo tanto no necesitan un gran despliegue de celdas solares para satisfacer sus necesidades energéticas.

- **Computadora de abordo.** Comúnmente conocida por las siglas OBC (*On Board Computer*), es el cerebro del satélite, se encarga de controlar todas las funciones del mismo, tanto como del módulo de comunicaciones, el sistema de energía, supervisión de las misiones y experimentos que se llevan a cabo en el satélite, etc. En general la computadora de abordo es un microprocesador, que va acompañado de una memoria flash *EEPROM*. Entre otras, las funciones de la computadora central son controlar que partes del satélite, deben ser energizadas en los momentos oportunos, interpretar las órdenes recibidas de tierra por el módulo de comunicaciones, supervisar el estado de

carga de las baterías, tomar información de los sensores de vuelo, mantener en progreso continuo el avance de la misión, etc.

- **Sistema de comunicaciones.** Como su nombre lo indica, es el encargado de actuar como interfaz entre el satélite y la base en tierra. Es de suma importancia ya que nos brinda información del estado del satélite y del desarrollo de las misiones del mismo. Generalmente cuenta con un *modem* y un transmisor, esta es una de las partes del satélite que más energía consume, ya que se necesita una gran cantidad de potencia para realizar transmisiones exitosas entre la tierra y el espacio.
- **Sistema de energía.** El Sistema Eléctrico de Poder abreviado *EPS* por sus siglas en inglés *Electric Power System*, es el encargado de alimentar a todos los sistemas del satélite, así como de recoger la energía solar a través de celdas fotovoltaicas, debe tener un adecuado banco de baterías así como un sistema de carga para las mismas y un sistema de conversión a los niveles de voltaje utilizados por los dispositivos instalados en el satélite, ya sean sensores, transmisores, microprocesadores, etc.
- **Misiones.** La tarea para la que se fabrican y se colocan en órbita satélites se le llama misión, las misiones de los satélites son bastante variadas, existen retransmisores para

telecomunicaciones, telescopios ópticos, satélites con cámaras en tiempo real para el estudio del clima, etc. A pesar de esto y por muy diferentes que las misiones sean, los satélites siempre cuentan con los mismos módulos principales mencionados. Todos estos equipos son considerados carga útil.

Para el Proyecto Wara se concluyó que se diseñara un nanosatélite tipo 3U, es decir, una estructura de 3 Cubesat, como se muestra en la **¡Error! No se encuentra el origen de la referencia..**

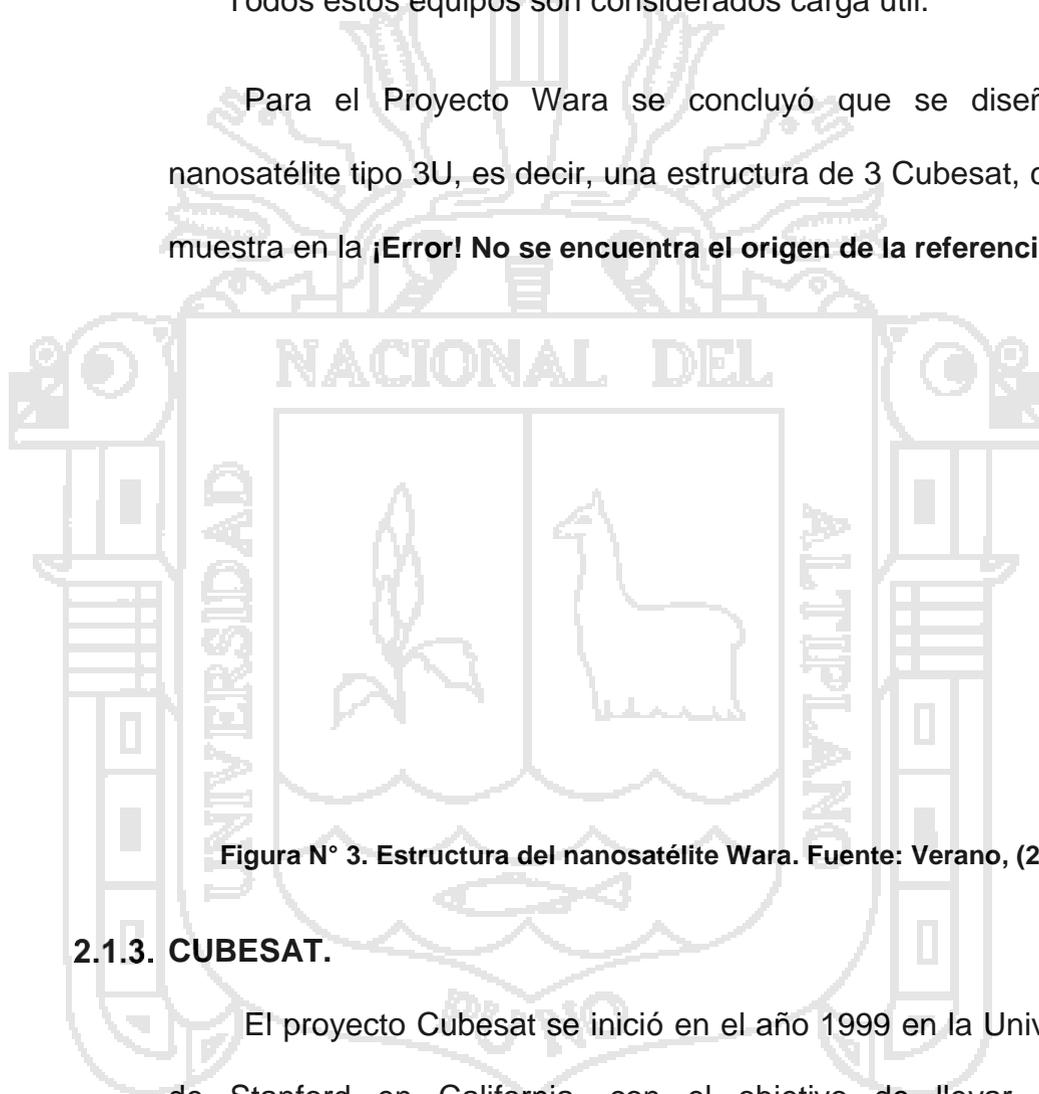


Figura N° 3. Estructura del nanosatélite Wara. Fuente: Verano, (2012).

2.1.3. CUBESAT.

El proyecto Cubesat se inició en el año 1999 en la Universidad de Stanford en California, con el objetivo de llevar a cabo investigación espacial a bajo costo y a nivel universitario. El estándar Cubesat especifica las características que estos satélites deben tener (CubeSat, 2013).

El Cubesat es un picosatélite de 10x10 cm. en su tamaño más común y un kilo de peso. El picosatélite lleva los componentes de un satélite de comunicaciones, no es un juguete a pesar de su apariencia y pequeño tamaño. Para su construcción se precisa mucho tiempo pues la programación de su CPU, el diseño de los sistemas de control, la fuente de alimentación, selección correcta de las baterías, compra de paneles solares, diseño del sistema de telemetría, transmisores y receptores espacio-tierra, fabricación del chasis e integración de todos los componentes requieren gran dedicación (Canarysat, 2011).

Su estructura se divide interiormente en bandejas las cuales llevan los distintos circuitos del satélite. Lo principal, la CPU será quien gobierne y controle todos los subsistemas. A esta unidad la podemos controlar desde la estación de tierra, de modo que podremos verificar si todos los subsistemas están funcionando correctamente o efectuar modificaciones en los parámetros del satélite que creamos necesarios. Llegado a este punto cabe resaltar un insignificante pero importante detalle cuando hablamos de reprogramar y de modificaciones, y es muy simple, se debe tener especial cuidado a la hora de cambiar parámetros en la programación que puedan afectar a los sistemas de comunicación, un error podría ser fatal pues por ejemplo la pérdida de comunicación entre el Cubesat y la estación de tierra supondría el

final del proyecto, hay que pensar que mientras se está testeando en tierra no habrá problema, otra cosa es ya cuando el satélite este en el espacio, en ese caso un error de programación salvo excepciones ya no tiene solución. Por ello es muy importante que las pruebas con la programación se efectúen en tierra y una vez salga del laboratorio este completamente testado.

En un Cubesat podemos distinguir diferentes partes:

- Paneles solares
- Reguladores de carga.
- Fuente de alimentación.
- Baterías.
- CPU.
- Radiomodem.
- Transmisor de radio.
- Receptor de radio.
- Giroscopio.
- Antenas.

A todo esto cabe añadir que las condiciones ambientales allá arriba no son las de la tierra, a las bajas temperaturas hay que añadir la radiación, por lo que el sistema deberá estar inmune a esas condiciones, un exceso de radiación acabaría con todos los circuitos del sistema.

Un Cubesat simple lleva solamente un baliza que puede transmitir bien analógico, en Morse o en digital utilizando el protocolo AX25, los Cubesat más complejos pueden llevar un sistema completo de transmisión y recepción vía radio. Las frecuencias de transmisión hacia la tierra suele ser en la banda de 70 cm., ósea, en las frecuencias de 430 a 440 MHz o bien en la banda de 2 metros de 145,850 a 146,000 MHz.

A fecha de hoy hay bastantes Cubesats en el espacio, varias universidades (América, Japón, Canadá, Colombia) han puesto en el espacio estos satélites, fruto de la colaboración entre los diferentes departamentos.

El costo total incluida la puesta en órbita esta entre los 40.000 y los 150.000 euros. Los costos pueden variar dependiendo de los sistemas que implementemos en el Cubesat y la variación constante de los precios de la puesta en el espacio.

2.1.4. PROYECTO QB50.

El proyecto QB50 se dio inicio el 27 de julio 2011, durante el Segundo Taller QB50 en VKI³,

El proyecto QB50 es un emprendimiento internacional, el cual espera colocar en la órbita baja terrestre (LEO) una red de cincuenta Nanosatélites, con sensores idénticos, con el objetivo principal de

³Von Karman Institute of Brussels, Belgium (Bélgica)

explorar la termosfera baja (Von Karman Institute and the European Space Agency, 2011).

La termosfera baja (90 a 320 km) es el menos explorado de la capa de la atmósfera. Exploradores de la atmósfera fueron trasladados en el pasado en órbitas muy elípticas (por lo general: 200 km de perigeo, 3000 km de apogeo), sino que lleva a los experimentos de mediciones in situ, pero el tiempo invertido en la región de interés por debajo de 320 km fue sólo unas pocas decenas de minutos. Mientras lo hacen explorar la termosfera entero inferior, el tiempo empleado en esta región es relativamente corto (unos minutos), sólo hay unos pocos vuelos al año y sólo proporcionan mediciones a lo largo de una sola columna. Poderosos instrumentos de teledetección a bordo de satélites de observación de la Tierra en órbitas más altas (600-800 km) recibir las señales de retrodispersión de componentes de la atmósfera a diferentes altitudes. Si bien esta es una excelente herramienta para explorar las capas inferiores de la atmósfera a unos 100 km, no es ideal para explorar la termosfera baja debido a que la atmósfera es tan enrarecida que la señal de retorno es débil. Lo mismo sucede con las observaciones de teledetección de la tierra con lidar y radares.

El multipunto, mediciones in situ de QB50 será complementaria a las observaciones de teledetección por los instrumentos en satélites de observación de la Tierra y las observaciones de

teledetección de la tierra con lidar y radares. Todos los modelos atmosféricos, y en última instancia, miles de usuarios de estos modelos, se beneficiarán de las medidas obtenidas por QB50 en la termosfera baja.

Una red de Cubesats en la termosfera baja en comparación con las redes en órbitas más altas tiene las siguientes ventajas:

- La vida útil de un Cubesat en el previsto en órbita terrestre baja sólo será de tres meses, es decir, mucho menos de los 25 años estipulados por las exigencias internacionales relacionados con los desechos espaciales.
- Una órbita terrestre baja permite altas tasas de datos debido a las distancias cortas de comunicación involucrados.
- En sus órbitas terrestres bajas del Cubesats estará por debajo de los cinturones de radiación de la Tierra, que es muy importante porque los componentes de Cubesats son de uso de bajo costo Commercial-Off-The-Shelf (COTS).

Los 50 Cubesats se pondrán en marcha a partir de Murmansk, en el norte de Rusia en una órbita circular a unos 320 km de altitud, la inclinación de 79°. Debido a la resistencia atmosférica, las órbitas de los Cubesats se deteriorarán y tendrán capas cada vez más bajas y más abajo de la termosfera se estudiará sin necesidad de propulsión a bordo, tal vez hasta 90 km. La altura de la órbita inicial

se seleccionará de manera que la vida de la misión de la Cubesats individuales será de unos tres meses. Para las universidades una misión de corta duración no debe un impedimento, el propósito para las universidades de desarrollar un proyecto Cubesat es principalmente educativo y los objetivos educativos que se pueden afrontar plenamente, incluso si la vida orbital es corto.

La intención es proporcionar de forma gratuita a los equipos participantes Cubesat el costo del lanzamiento, el sistema de despliegue de diseño personalizado, los sensores estándar para la investigación atmosférica, los servicios de lanzamiento y las interfaces con las autoridades del vehículo de lanzamiento, pruebas ambientales (si se solicita), el transporte de los 50 Cubesats al lugar de lanzamiento y las pruebas caja Cubesat durante la campaña de lanzamiento.

Los equipos participantes Cubesat son libres de usar cualquier espacio que queda en la unidad “funcional” de la Cubesat doble, para un paquete de tecnología o un sensor de su propia elección. Se espera que los equipos presenten la documentación requerida en el momento oportuno, participar en las evaluaciones de proyectos importantes, el apoyo a la campaña de ensayos ambientales (pero no la campaña de lanzamiento), operen sus Cubesat en órbita y faciliten sus datos de la ciencia y los datos limitados de limpieza

dentro de los 6 – 9 meses después del lanzamiento de la QB50 de procesamiento del centro de datos y archivo.

Los sensores serán idénticos para todos los 50 Cubesats. 38 Cubesats se prevé que se impartirán en las universidades en 22 países europeos, ocho por las universidades en los EE.UU., dos de las universidades en Canadá y dos por las universidades japonesas. También se ha expresado Interés en participar en QB50 en más de 20 universidades en Australia, Brasil, Chile, Egipto, Israel, Perú, Puerto Rico, Rusia, Sudáfrica, Corea del Sur, Taiwán, Turquía y Vietnam.

Varias Universidades del Perú firmaron un Consorcio en el Proyecto QB50⁴, donde se encontraba liderando la Universidad Nacional del Altiplano, a través de la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas.

2.1.5. ESTACIÓN TERRENA ESPACIAL.

El término “estación terrena” es utilizado indistintamente para indicar todo equipo terminal que se comunica desde la Tierra con un satélite, sin importar si esta fijo en algún punto, o si está instalado en un barco, avión, o cualquier vehículo terrestre como un autobús, un auto o un camión de carga (Neri Vela, 2003).

⁴Peruvian Consortium: University of the Highland Puno, University San Antonio Abad of Cusco, University San Agustín of Arequipa, and Partners; PERU.

Una estación terrena espacial es un conjunto de comunicaciones y de computo que puede ser terrestre (fijo y móvil). Marítimo o aeronáutico. Las estaciones terrenas pueden ser usadas en forma general para transmitir y recibir del Satélite.

El tamaño y complejidad de una estación terrestre depende del servicio que será provisto y la potencia radiada por el satélite. Las estaciones más simples permiten solo recepción y están equipadas con una antena parabólica que puede tener un diámetro de menos de 1 m., mientras que otras estaciones usan antenas mucho más grandes.

Se ilustra el diagrama general de una estación terrena; pero, dependiendo de su aplicación particular, pueden carecer de uno o varios bloques indicados. Por sus características de radiación, todas las estaciones terrenas que tengan antenas pequeñas no necesitan sistema de rastreo, mientras que las de diámetros muy grande, como las de comunicaciones internacionales, si lo requieren para conservar su angosto haz directivo bien apuntado hacia el satélite.

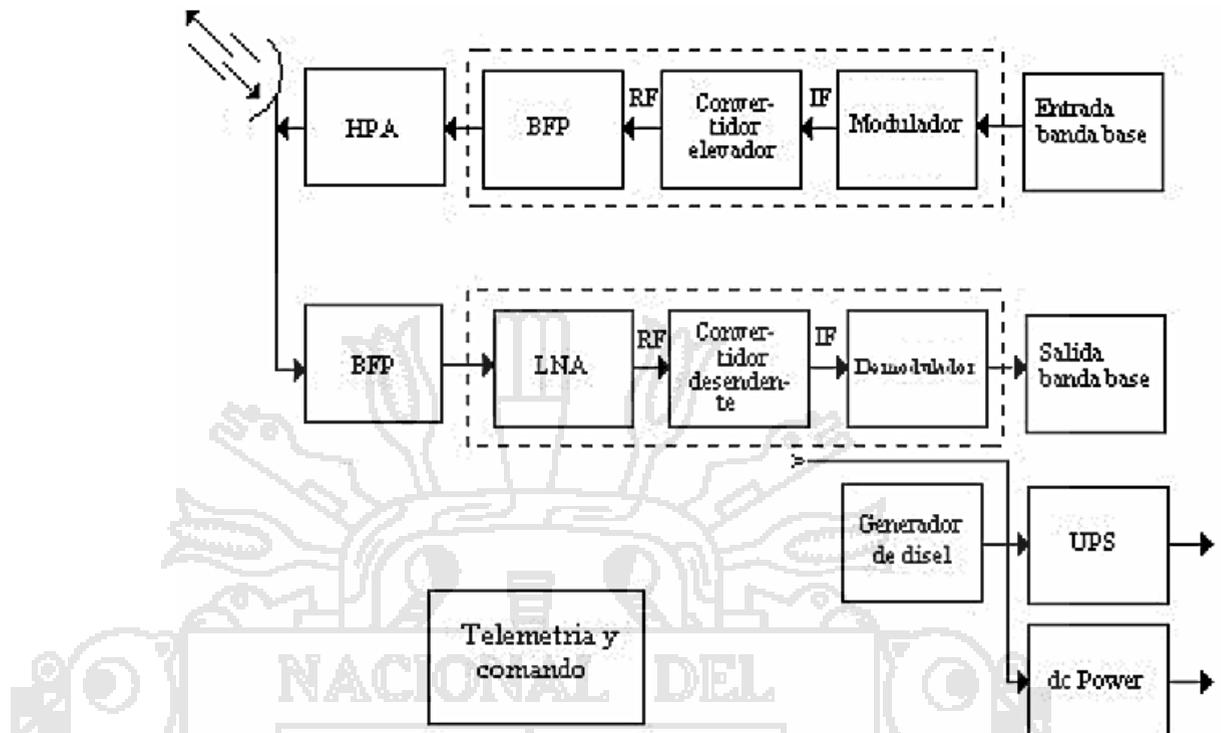


Figura N° 4. Diagrama general de una estación terrena común. Fuente: Peredo, (2004).

2.1.6. TELEMETRÍA Y TELECOMANDO SATELITAL.

TELEMETRÍA.

Según el diccionario de la lengua española la telemetría es una tecnología que permite la medición remota de magnitudes físicas y el posterior envío de la información hacia el operador del sistema.

Algunas empresas describen que el proceso de telemetría consiste en la transmisión de datos desde un sensor a un servidor donde los datos son almacenados para su administración y visualización, usando modernos sistemas de gestión diseñados a medida de las necesidades de cada cliente; que permiten

visualización on-line de eventos y alarmas, generación de reportes estadísticos, almacenamiento de datos históricos, etc.

El envío de información hacia el operador en un sistema de telemetría se realiza típicamente mediante comunicación inalámbrica, aunque también se puede realizar por otros medios (teléfono, redes de ordenadores, enlace de fibra óptica, etcétera). Los sistemas de telemetría reciben las instrucciones y los datos necesarios para operar mediante desde el Centro de Control.

La telemetría se utiliza en grandes sistemas, tales como naves espaciales, plantas químicas, redes de suministro eléctrico, redes de suministro de gas entre otras empresas de provisión de servicios públicos, debido a que facilita la monitorización automática y el registro de las mediciones, así como el envío de alertas o alarmas al centro de control, con el fin de que el funcionamiento sea seguro y eficiente. La Telemetría se utiliza en infinidad de campos, tales como la exploración científica con naves tripuladas o no (submarinos, aviones de reconocimiento y satélites), diversos tipos de competición (por ejemplo, Fórmula 1 y MotoGP), o la operación de modelos matemáticos destinados a dar sustento a la operación de embalses.

En las fábricas, oficinas y residencias, el monitoreo del uso de energía de cada sección o equipo y los fenómenos derivados (como la temperatura).

TELECONTROL.

El telecontrol o telemando consiste en el envío de indicaciones a distancia mediante un enlace de transmisión utilizando órdenes enviadas para controlar un sistema o sistemas remotos que no están directamente conectados al lugar desde donde se envía el telecontrol. El telecontrol se puede llevar a cabo en tiempo real o no dependiendo de las circunstancias (Grupo Hispasat, 2014).

El proceso telemétrico consiste en el uso de actuadores con capacidad de ejecutar acciones tales como encender/apagar interruptores, bloquear/desbloquear dispositivos de seguridad, abrir/cerrar válvulas, aumentar/disminuir flujos de fluidos, intensidad de velocidad, intensidad de voltaje etc. Para evitar el acceso no autorizado al sistema remoto, se puede emplear algún tipo de cifrado o un secreto de clave compartida.

El Telecontrol se emplea desde el control desde el sofá de: un televisor, el aire acondicionado, las luces, las ventanas, etc.; hasta las operaciones de: Guiado a distancia de armas o misiles, hacer volar un avión controlado por radio, entre otros.

Los sistemas que necesitan medición remota y reporte de información de interés para el diseñador del sistema o el operador deben usar la contrapartida del telecontrol, la telemetría.

Así por ejemplo se aplica, tanto la telemetría y el telecontrol, para administrar la temperatura de tanques y cámaras frigoríficas, caudal de flujo en ductos, ciclos de operación en grupos electrógenos, datos críticos en maquinarias, temperatura, presión de aire/aceite, carga de combustible; niveles de capacidad en tanques y contenedores, entre más aplicaciones. Un ejemplo más relativo al tema de investigación es el empleado por las agencias espaciales como la NASA, la UK Space Agency, la ESA y otras, las cuales utilizan sistemas de telemetría y de telecontrol para operar con naves espaciales y satélites (Deutsche Welle, 2014).

Finalmente, la Telemetría y Telecomando es la función básica de todo por más simple que sea la nave espacial se requiere un amplio contacto con las estaciones terrestres de control, comando y retorno de datos; en una nave espacial su sistema de comando y gestión de datos se encarga de enviar y recibir todos los datos. Incluidos los datos científicos y las operaciones de la nave o la carga útil. (European Space Agency [ESA], 2014).

2.2. MARCO CONCEPTUAL.

2.2.1. CUBESAT.

Es un picosatélite de 10x10 cm. en su tamaño más común y un kilo de peso, este lleva los componentes de un satélite de comunicaciones. Su estructura está compuesta por distintos

circuitos; lo principal, la CPU será quien gobierne y controle todos los subsistemas. (Canarysat, 2011).

2.2.2. DATO CRUDO.

Los datos crudos es un término para los datos que se recogen desde una fuente, que no han sido sujetos al procesamiento u otra manipulación. Aunque los datos crudos tienen el potencial de volverse "información", requiere una extracción selectiva, de organización y a veces de análisis y formateo para su presentación. (Computer Hope, 2014).

2.2.3. ESTACIÓN TERRENA ESPACIAL.

El término es utilizado indistintamente para indicar todo equipo terminal que se comunica desde la Tierra con un satélite, sin importar si esta fijo en algún punto o si está instalado en un barco, avión, o cualquier vehículo terrestre como un autobús, un auto o un camión de carga. (Neri Vela, 2003).

2.2.4. MODELO EN CASCADA.

También llamado Modelo Lineal Secuencial, o como ciclo de vida del software. Las etapas de este modelo se transforman en actividades fundamentales de desarrollo: Análisis y definición de requerimientos, Diseño del sistema y del software, Implementación y prueba de unidades, integración y Prueba del sistema, funcionamiento y Mantenimiento. (Sommerville, 2005).

2.2.5. NANOSATÉLITE.

Es un satélite de clasificación según su masa de 1 a 10 Kg. Los satélites artificiales en general cuentan con las mismas partes principales, ya sean para misiones de exploración temporales, o para misiones de larga duración, las partes más importantes de un Nanosatélite son: estructura mecánica, computadora de abordo, sistema de comunicaciones, sistema de energía y las Misiones. Mozombite, (2012)

2.2.6. PROYECTO QB50.

El proyecto QB50 es un emprendimiento internacional, el cual espera colocar en la órbita baja terrestre (LEO) una red de cincuenta Nanosatélites, con sensores idénticos, con el objetivo principal de explorar la termosfera baja. Los 50 satélites serán desarrollados por distintas Universidades del mundo. (VKIEA, 2011).

2.2.7. SATÉLITES ARTIFICIALES.

Los satélites artificiales son máquinas hechas por el hombre, estas máquinas son lanzadas al espacio y orbitan la Tierra u otro cuerpo en el espacio con un fin. (NASA, 2011).

2.2.8. SOFTWARE.

Programas de ordenador, la documentación asociada y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta. (Sommerville, 2005).

2.2.9. TELECONTROL.

El telecontrol o telemando consiste en el envío de indicaciones a distancia mediante un enlace de transmisión utilizando órdenes enviadas para controlar un sistema o sistemas remotos que no están directamente conectados al lugar desde donde se envía el telecontrol. (Hispasat, 2014).

2.2.10. TELEMETRÍA.

La telemetría es una tecnología que permite la medición remota de magnitudes físicas y el posterior envío de la información hacia el operador del sistema, envío que se realiza habitualmente mediante comunicación inalámbrica. (Hispasat, 2014).

2.3. HIPÓTESIS DE LA INVESTIGACIÓN.

El uso del software de análisis, interpretación y control en la Estación Terrena, optimiza los procesos de Telemetría y Telecontrol del nanosatélite Wara.

2.3.1. OPERACIONALIZACIÓN DE VARIABLES.

Variable	Dimensiones	Indicadores	Escala
(Independiente) Software de análisis, interpretación y programación de comandos.	Factores de calidad de Software según ISO 9126	Funcionalidad	
		Fiabilidad	
		Usabilidad	
		Eficiencia	
		Mantenibilidad	
(Dependiente) Optimización de los procesos de Telemedición y Telecontrol del nanosatélite Wara.	Proceso de Telemedición	Facilidad de análisis e interpretación	Likert (sencillo – difícil)
		Rapidez de interpretación	Likert (rápido – lento)
		Fiabilidad de la interpretación	Likert (afirmación rotunda – negación rotunda)
	Proceso de Telecontrol	Entendimiento gráfico para el análisis.	Likert (completamente – nada)
		Facilidad de programar comandos.	Likert (sencillo – difícil)
		Fiabilidad de programar comandos.	Likert (afirmación rotunda – negación rotunda)
	Manejabilidad gráfica para programar los comandos.	Likert (completamente – nada)	

Tabla N° 2. Operacionalización de variables. Fuente: elaboración Propia.

CAPÍTULO III : MÉTODO DE INVESTIGACIÓN



3.1. TIPO Y DISEÑO DE INVESTIGACIÓN.

Tipo de Investigación.

El tipo de investigación corresponde al enfoque Cuantitativo, y de alcance correlacional.

Diseño de Investigación.

El diseño en uso será Experimental, y en este caso se considera de tipología Cuasiexperimental, debido a que el grupo de estudio es un grupo sólido y ya establecido, como son los docentes integrantes del Proyecto Wara, bajo el diseño pre prueba/post prueba con un solo grupo:

G:	0 ₁	X	0 ₂
----	----------------	---	----------------

Dónde:

- G: Grupo Experimental.
- 0₁: Pre prueba.
- X: Tratamiento con la variable independiente (Software).
- 0₂: Post prueba.

3.2. POBLACIÓN Y MUESTRA.

Población.

La población la conforman todos los participantes en el proyecto Wara de la FIMEES, así como de los ingenieros en el extranjero del proyecto QB50, así se tendría:

- Docentes y estudiantes integrantes del Proyecto Wara en la FIMEES.
- Profesionales en el extranjero integrantes del Proyecto QB50.

Muestra.

Para la muestra, como se mencionó que el diseño será cuasiexperimental, y según el interés principal del proyecto la muestra es no probabilística, por lo cual se consideraran a los Profesionales Integrantes del Proyecto QB50 peruanos en el extranjero que actúan como consultores para el proyecto Wara, los Ingenieros Juan Martin Canales Romero, Jaime Estela Gutiérrez y Avid Román Gonzales, quienes evaluarán el desempeño del software. Así mismo, los Docentes Integrantes de la Comisión de la ejecución del Proyecto QB50 en la FIMEES.

Usuarios	Muestra
Ingenieros Consultores en el Extranjero	3
Docentes FIMEES	5
TOTAL	8

Tabla N° 3. Muestra para la investigación. Fuente: elaboración propia.

3.3. MÉTODO DE RECOPIACIÓN DE DATOS.

El método utilizado para la recolección de datos, fue mediante **Cuestionarios**. Ya que dicho instrumento de medición nos permitió plantear varias preguntas cerradas, para poder obtener la información necesaria y fueron debidamente sistematizadas para su análisis correspondiente.

Los cuestionarios fueron aplicados de manera autoadministrada, es decir, se proporcionó directamente a la muestra en contextos individual

para los docentes de la FIMEES y mediante envió (correo electrónico) a los ingenieros consultores en el Extranjero.

3.4. MÉTODO DE PROCESAMIENTO DE DATOS.

Se realizó el procesamiento en matrices de datos, para evaluar la confiabilidad de los resultados obtenidos de las encuestas se empleó el coeficiente Alfa de Cronbach, utilizando software de cálculos y análisis estadístico. Dónde: si obtenemos 0.25 en el coeficiente, esto indica baja confiabilidad; si el resultado es 0.50, la fiabilidad es media o regular. Si supera el 0.75 es aceptable, y si es mayor a 0.90 es elevada.

3.5. MÉTODO DE ANÁLISIS DE DATOS.

Para el análisis de los datos se utilizó el Software estadístico SPSS® versión 20, y se utilizó la prueba *t*, *el cual* se basa en una distribución muestral o poblacional de diferencia de medias conocida como la distribución *t* de Student que se identifica por los grados de libertad, los cuales constituyen el número de maneras en que los datos pueden variar libremente.

Empleamos la prueba *t* de Student para demostrar si existe diferencia significativa entre la Pre prueba y la Post prueba, claro después de aplicar el tratamiento (software).

CAPÍTULO IV : RESULTADOS Y DISCUSIÓN



4.1. ESPECIFICACIÓN DEL SOFTWARE.

La especificación del software o ingeniería de requerimientos es el proceso de comprensión y definición de qué servicios se requieren del sistema y de identificación de las restricciones de funcionamiento y desarrollo del mismo. (Sommerville, 2005, pág. 69).

Para la captura de requisitos se utilizó las siguientes técnicas: Entrevistas y estudio de la documentación. Se realizó entrevistas vía web con los ingenieros asesores del proyecto Wara en el extranjero, ingenieros Jaime Estela y Avid Roman. Así mismo, se revisó toda la documentación existente para el Proyecto Wara.

4.1.1. DESCRIPCIÓN GENERAL.

4.1.1.1. Perspectiva del Producto.

El software será parte de la Estación Terrena Espacial, y funcionara conjuntamente con el Modulo de comunicación Satelital. Dicho modulo estudiara la forma de comunicación con el nanosatélite y su objetivo es establecer la comunicación adecuada y garantizar que la información enviada llegue correctamente y viceversa. Se muestra en la Figura N° 5, el contexto del software.



Figura N° 5. Esquema de la estación terrena espacial – WARA. Fuente: elaboración propia.

4.1.1.2. Funcionalidad del Producto.

El Software tendrá dos funciones únicas e importantes dentro de la Estación Terrena Espacial, la de interpretar y medir la información recibida del nanosatélite, y de poder realizar comandos e instrucciones para ser enviados al nanosatélite.

4.1.1.3. Características de los Usuarios.

Dentro de La Estación Terrena Espacial estará personal dedicado del Proyecto WARA, y sabemos que todos los integrantes del proyecto son tanto decentes, como alumnos de la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y

Sistemas. Por lo que podemos concluir que el personal tiene el siguiente perfil:

Tipo de usuario	Encargados de Telemedición y Telecontrol
Formación	Nivel Superior
Habilidades	Sistemas de Información, Programación, Electrónica, Mecánica Eléctrica, Telemática.
Actividades	Estudiar la medición y realizar comandos mediante el software.

Tabla N° 4. Características de los usuarios. Fuente: elaboración Propia.

Se podría mencionar que quienes estén en esta tarea de la Telemedición y Telecontrol, estarán tecnológicamente bien capacitados.

4.1.1.4. Restricciones.

Restricciones de Hardware.

El software debe de funcionar en una computadora estándar Pentium IV como mínimo de procesador. Y su espacio en Disco Duro debe ser menor a 10 Mb.

Restricciones de Sistema Operativo.

El software tendrá que ser multiplataforma, es decir, que debe de poderse instalar en equipos con Sistema Operativo Microsoft Windows o en cualquier otro de Software Libre.

Restricciones de Lenguaje de Programación.

El lenguaje de programación en el que se implemente el software, será uno que pueda ser ejecutado tanto en Microsoft Windows o en otro de Software Libre.

Restricciones del Software Propiamente.

La principal restricción que tendrá el software; es que dependerá absolutamente de la programación del Microcontrolador que hará de procesador del Nanosatélite.

Al programar el microcontrolador con los sensores, se generara un formato de recopilación de datos, los cuales serán enviados a la Estación Terrena Espacial, este formato tiene que saber interpretarlo el Software. Del mismo modo, para cuando se realice las instrucciones en Tierra y al ser enviados, el microcontrolador sepa ejecutar dichas instrucciones.

4.1.1.5. Suposiciones y Dependencias.

Para el desarrollo y ejecución del software se tiene la siguiente suposición: Que ya se tiene comunicación con el nanosatélite, con envío y recepción de archivos satisfactoriamente.

4.1.2. REQUISITOS ESPECÍFICOS.

4.1.2.1. Requisitos comunes de Interfaces.

Interfaces de Usuario.

La principal característica de la interfaz de usuario es que esta debe contar con gráficos explícitos para poder interpretar y estudiar los datos, tanto para las mediciones y para realizar los comandos, especialmente en comandos.

Interfaces de Hardware.

En lo que corresponde al Hardware para el uso del software, se utilizara básicamente teclado y mouse como dispositivos de entrada, y un monitor o monitores e impresora opcionalmente.

Interfaces de Software.

Nombre del Software	Java
Descripción	Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90.
Propósito	Implementación del software.
Definición de Interfaces	Trabaja con las extensiones *.jar y *.java

Tabla N° 5. Interfaz de software: Java. Fuente: elaboración propia.

Interfaces de Comunicación.

Como el Software estará instalado dentro de la Estación Terrena Espacial, posiblemente trabaje junto a otros ordenadores y se requiera compartir los datos. Por lo que el equipo donde se instale el Software debe utilizar el protocolo estándar de red de área local para compartir la información.

Nombre	Protocolo TCP/IP.
Descripción	Es el protocolo usado para las conexiones de la red local
Propósito	Permitirá la comunicación entre los diferentes ordenadores.

Tabla N° 6. Interfaz de comunicación: protocolo TCP/IP. Fuente: Elaboración Propia.

4.1.2.2. Requisitos Funcionales.

Los requisitos funcionales fueron obtenidos en base a los subsistemas también conocidos como módulos que componen el nanosatélite, principalmente para monitorear su funcionamiento. Junto con ellos demás requisitos que debiera cumplir el software.

Requisito Funcional 1.

Identificador	RF1
Nombre	Grafica funcionalidad de voltaje y corriente.
Prioridad	Alta.
Descripción	<p>El Software mostrara las lecturas de voltaje y corriente que tiene el nanosatélite de forma gráfica, con respecto al tiempo.</p> <p>Entrada: Archivo de Datos.</p> <p>Proceso: Graficar en cuadro estadístico.</p> <p>Salida: Muestra un Gráfico del estado de voltaje y corriente con todas las lecturas realizadas hasta la última descarga.</p>
Suposición	Existen ya datos en el Archivo de Datos.

Tabla N° 7. Requisito funcional 1. Fuente: elaboración propia.

Requisito Funcional 2.

Identificador	RF2
Nombre	Grafica funcionalidad de Temperaturas Internas
Prioridad	Alta.
Descripción	<p>El Software mostrara las lecturas de las temperaturas de los componentes principales del nanosatélite (Procesador, Cubo 1, Cubo 2 y Cubo3) de forma gráfica.</p> <p>Entrada: Archivo de Datos.</p> <p>Proceso: Graficar en cuadro estadístico.</p> <p>Salida: Muestra un Gráfico del estado de temperaturas interiores con todas las lecturas realizadas hasta la última descarga.</p>
Suposición	Existen ya datos en el Archivo de Datos.

Tabla N° 8. Requisito funcional 2. Fuente: elaboración propia.

Requisito Funcional 3.

Identificador	RF3
Nombre	Grafica la funcionalidad de Generadores de Energía Eléctrica.
Prioridad	Alta.
Descripción	<p>El Software mostrara las lecturas del funcionamiento de los generadores de energía eléctrica del nanosatélite (Paneles Solares, Dinamo y Termo-couplas) de forma gráfica.</p> <p>Entrada: Archivo de Datos.</p> <p>Proceso: Graficar en cuadro estadístico.</p> <p>Salida: Muestra un Gráfico del funcionamiento de los generadores de energía eléctrica con todas las lecturas realizadas hasta la última descarga.</p>
Suposición	Existen ya datos en el Archivo de Datos.

Tabla N° 9. Requisito funcional 3. Fuente: elaboración propia.

Requisito Funcional 4.

Identificador	RF4
Nombre	Grafica posicionamiento del nanosatélite.
Prioridad	Alta.
Descripción	<p>El Software mostrara las lecturas del posicionamiento del nanosatélite (respecto a sus tres ejes de orientación) de forma gráfica.</p> <p>Entrada: Archivo de Datos.</p> <p>Proceso: Graficar en cuadro estadístico.</p> <p>Salida: Muestra un Gráfico de las coordenadas del nanosatélite con todas las lecturas realizadas hasta la última descarga.</p>
Suposición	Existen ya datos en el Archivo de Datos.

Tabla N° 10. Requisito funcional 4. Fuente: elaboración propia.

Requisito Funcional 5.

Identificador	RF5
Nombre	Regulación del voltaje y corriente.
Prioridad	Alta.
Descripción	<p>El Software tendrá la opción de poder regular el voltaje y corriente del nanosatélite.</p> <p>Entrada: Graduar un valor para voltaje y/o corriente.</p> <p>Proceso: Empaquetar comando.</p> <p>Salida: Se genera un archivo de comandos para ser enviado al Nanosatélite.</p>
Suposición	Los datos graduados son adecuados para reconfiguración del nanosatélite.

Tabla N° 11. Requisito funcional 5. Fuente: elaboración propia.

Requisito Funcional 6.

Identificador	RF6
Nombre	Regulación de las temperaturas interiores.
Prioridad	Alta.
Descripción	<p>El Software tendrá la opción de poder regular las temperaturas de los componentes internos del nanosatélite.</p> <p>Entrada: Graduar un valor para temperatura.</p> <p>Proceso: Empaquetar comando.</p> <p>Salida: Se genera un archivo de comandos para ser enviado al Nanosatélite.</p>
Suposición	Los datos graduados son adecuados para reconfiguración del nanosatélite.

Tabla N° 12. Requisito funcional 6. Fuente: elaboración propia.

Requisito Funcional 7.

Identificador	RF7
Nombre	Regulación del posicionamiento del Nanosatélite.
Prioridad	Alta.
Descripción	<p>El Software tendrá la opción de poder regular la posición del nanosatélite, para la orientación correcta.</p> <p>Entrada: Graduar valores en eje X, Y y Z.</p> <p>Proceso: Empaquetar comando.</p> <p>Salida: Se genera un archivo de comandos para ser enviado al Nanosatélite.</p>
Suposición	Los datos graduados son adecuados para reconfiguración del nanosatélite.

Tabla N° 13. Requisito funcional 7. Fuente: elaboración propia.

Requisito Funcional 8.

Identificador	RF8
Nombre	Interpretación de datos crudos obtenidos del Nanosatélite.
Prioridad	Muy Alta.
Descripción	<p>El Software tendrá la opción de cargar la información recibida en Tierra para interpretarlo y almacenarla para su gestión de datos.</p> <p>Entrada: Un archivo.</p> <p>Proceso: Cargar el archivo, interpretarlo y adjuntarlo a sus datos.</p> <p>Salida: Almacenamiento de los datos interpretados en la base de datos.</p>
Suposición	El archivo es el recibido por el nanosatélite, es de formato correcto y los datos son nuevos.

Tabla N° 14. Requisito funcional 8. Fuente: elaboración propia.

Requisito Funcional 9.

Identificador	RF9
Nombre	Grafica funcionalidad de Temperaturas Exteriores.
Prioridad	Opcional.
Descripción	El Software mostrara las lecturas de las temperaturas del exterior del nanosatélite de forma gráfica. Entrada: Archivo de Datos. Proceso: Graficar en cuadro estadístico. Salida: Muestra un Gráfico del estado de temperaturas exteriores con todas las lecturas realizadas hasta la última descarga.
Suposición	Existen ya datos en el Archivo de Datos.

Tabla N° 15. Requisito funcional 9. Fuente: elaboración propia.

Requisito Funcional 10.

Identificador	RF10
Nombre	Acceso al software.
Prioridad	Alta.
Descripción	El Software deberá tener un acceso con una identificación para su empleo. Entrada: variable de Identificación. Proceso: Autenticar variable con almacenada. Salida: Permite acceso al software para su uso y manejo.
Suposición	Existen ya datos en el Archivo de Datos.

Tabla N° 16. Requisito funcional 10. Fuente: elaboración propia.

4.1.2.3. Requisitos No Funcionales.

Requisitos de Rendimiento.

El software deberá de realizar todos los procesos funcionales en menos de 3 segundos, así mismo, debe de tener soporte para visualizar hasta 20 ventanas simultaneas.

Seguridad.

Contará con acceso únicamente a los integrantes del proyecto WARA, en base a una identificación y/o contraseña que estará bajo encriptación; Específicamente será para los responsables de la monitorización del nanosatélite.

Fiabilidad.

- Se espera que el software no tenga fallo alguno, pero se consentirá un máximo de 2 fallos por mes, en caso extremo.
- El software debe tener opciones de reconfiguración y recuperación en caso de fallos, ya sea de funcionamiento o de fuente de datos. Dichas opciones deben ser fáciles y sencillos de ejecutar.
- En lo posible el software tendrá que recuperarse automáticamente en caso de fallos, en su defecto este deberá indicar.

Disponibilidad.

El software deberá estar 100% disponible para su uso.

Mantenibilidad.

- En cuanto a estabilidad, el software deberá garantizar estabilidad de uso. Si en algún momento futuro necesite una reprogramación o actualización, esta debe ser realizada en otra versión nueva, sin intervenir en el actual software.
- El uso del software será en el tiempo del experimento en que el Nanosatélite este en órbita, el cual será desde 3 a un máximo de 6 meses. Por lo cual será necesario dar un único mantenimiento o mejor dicho una actualización ya sea de funciones, de cálculos u otra necesidad, antes de comenzar el experimento.
- El mantenimiento o actualización deberá ser dado por un experto en programación en base a las nuevas especificaciones que se requieran.

Portabilidad.

- El software tendrá un ejecutable o un instalador único, que incluya todo los componentes, complementos y base de datos.
- El software estará implementado en un lenguaje de programación multiplataforma.

- El software se podrá ejecutar en un cualquier Sistema Operativo, tanto en Software Privativo o de Software Libre.

4.2. ANÁLISIS Y DISEÑO DEL PROCEDIMIENTO DE INTERPRETACIÓN DE DATOS CRUDOS Y LA PROGRAMACIÓN DE COMANDOS.

¿Por qué realizar un punto aparte para el análisis y diseño del procedimiento de interpretación de datos crudo y la programación de comandos? Porque es una de las funciones fundamentales que realizara el software, es la base para la ejecución de los demás requisitos funcionales.

A partir de una interpretación correcta y exacta, se podrán realizar los gráficos, medidores y reguladores que servirán para mejorar el monitoreo del nanosatélite.

4.2.1. PROCEDIMIENTO DE INTERPRETACIÓN DE DATOS CRUDOS.

Como se mencionó en los puntos anteriores, el tiempo de comunicación activa con el nanosatélite es tan solo de 10 minutos en promedio, en este tiempo se recibirán los archivos del nanosatélite que los llamaremos archivos *lecturas*.

LOS ARCHIVOS.

El nanosatélite empaquetara todo lo que captan sus sensores, como dijimos: temperaturas, voltaje y corriente, energía eléctrica, estado de CPU, entre otros. Todos estos datos capturados serán empaquetados en un archivo lectura, en forma codificada.

LOS DATAGRAMAS.

Es el conjunto de datos crudos, los llamamos datos crudos porque están en forma codificada, todo ese conjunto será denominado datagrama

Un archivo lectura, contendrá un datagrama y tiene la siguiente estructura básica:

```

39 17 34 17 15 37 30 17 32 15 25
000001261213 0121 099
11 32 17 30 21 28 35 31 28 17 34 13
000201251213 020321251213 2113
120104251213 135939251213 2129
11 36 21 34 29 31 19 31 37 32 28 17 13
000000251213 114 114 115 116 119 113 113 2104
010000251213 119 112 116 114 116 118 118 2113
11 20 25 30 17 29 31 13
000000241213 04279 34019
010000241213 04277 34002
020000241213 04285 340221
11 36 21 29 32 25 30 36 21 34 25 31 34 13
000000241213 010 026 012 023
001000241213 015 029 010 027
11 36 21 29 32 21 40 36 21 34 25 31 34 13
000000241213 0127
001000241213 0110
002000241213 0110
...

```

Figura N° 6. Datagrama o datos crudos recibidos del nanosatélite Wara.
Fuente: elaboración propia.

Con una estructura así, la interpretación de estos datos, resulta complejo, demoroso y con riesgo a equivocarse en la interpretación. Entonces para el desarrollo del software se tiene que instruirle para que logre interpretar estos datagramas y mostrarlos de una manera

mucho más entendible, para facilitar el análisis de los datos que nos manda el nanosatélite.

Pasamos a explicar la estructura del datagrama:

A nivel de fila existen dos tipos de dato que los llamamos: nombre y registro.

1) Nombre. Se distingue por que está compuesto por números que van de 2 dígitos separados de un espacio, por ejemplo:

39 36 21 34 29 31 19 31 37 32 28 17 13
 W C

Cada conjunto de 2 dígitos representan un carácter que para su fácil almacenamiento se le resto 80 al valor decimal, ejemplos en la Tabla N° IV.13.

Carácter	Decimal	Representación (valor decimal - 80)
W	119	39
R	114	34
A	97	17

Tabla N° 17. Representación de caracteres en su valor decimal - 80. Fuente: elaboración propia.

Este tipo de fila, aparece para distinguir cada módulo que se está midiendo, representa el nombre de programación de dichos módulos. Por ejemplo: [voltajeycorriente], [panelsolar], [posicion].

2) Registro. Son todos los demás valores que no son nombre, por ejemplo:

160049251213 180533251213 2101

010000251213 119 112 116 114 116 118 118 2113

Estos registros almacenan dos tipos de datos: hora-fechas y valores numéricos. Dentro de los registros, la hora-fecha son todas las subcadenas de 12 dígitos, de los ejemplos:

160049251213 180533251213 2101

hora-fecha

De la fila anterior se puede observar que existe dos subcadenas que representan hora y fecha. Su interpretación es tal y como se extiende la subcadena:

160049251213

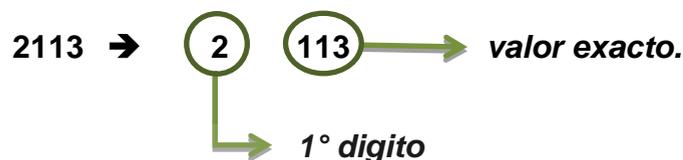
16:00:49 25/12/13

Los valores numéricos en los registros son los demás subcadenas de menos de 12 dígitos, de los ejemplos:

010000251213 119 1920 116 114 2113

Valores numéricos

Estas subcadenas representan valores y variables, para interpretarlas se desfragmenta en dos partes:



El valor del 1° digito es la cantidad de dígitos que se consideran enteros en el valor exacto, comenzando por la izquierda.

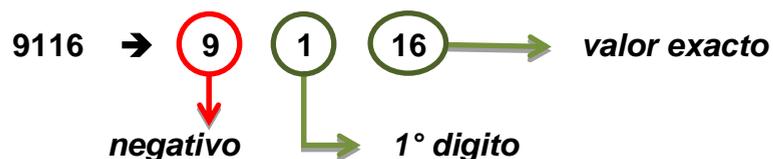
Del ejemplo: 1,1,3 dos dígitos enteros.

Obtenemos: **11.3**

En algunos casos se presentaran valores negativos, pero ¿cómo se distinguen? Estos son antecidos por un 9, un ejemplo sería cuando encontremos valores numéricos así:



Entonces si encontramos valores numéricos con un 9 al inicio de subcadena, esto nos indicaría que se trata de un valor negativo. Entonces hacemos el mismo procedimiento de interpretación para valores numéricos, excluyendo al 9, Por ejemplo:



Vemos que el valor exacto es 16 y el 1° dígito es 1, entonces quiere decir que un solo dígito es entero, y el nueve al comienzo hace que el resultado sea un número negativo.

Del ejemplo: $\overset{\curvearrowright}{1}6$ un dígito entero.

Obtenemos: -1.6

Y de esta manera se interpreta los datagramas, y el software tiene que realizar todo el procedimiento para interpretar todas las lecturas que se recibirá en la Tierra.

4.2.2. PROCEDIMIENTO PARA LA PROGRAMACIÓN DE COMANDOS.

La programación de los comandos tiene una lógica similar al procedimiento de interpretación, pero de manera inversa. Esta vez, se tiene que empaquetar los comandos en un formato que entienda el nanosatélite, la mejor manera es en dato crudo también.

El software mostrará los controles para hacer las órdenes respectivas al nanosatélite, y al momento de empaquetar dichas órdenes, estas se guardará en un archivo comando, que recogerá todos los cambios y regulaciones hechas bajo las siguientes especificaciones:

Según de que sensor se haya realizado la regulación, se guardará con el nombre de módulo en formato decimal - 80. Por ejemplo uno de los módulos que mide las temperaturas interiores

para el correcto funcionamiento de los componentes del nanosatélite, es nombrado *templInterior*. Entonces:

t e m p i n t e r i o r

Se guarda: 36 21 29 32 25 30 36 21 34 25 31 34

En la siguiente línea irán los valores regulados para este módulo, separados por espacios, y claro codificados.

012 013 012 0120

El software entonces deberá de empaquetar los comandos solo de los módulos regulados o modificados, y generar un archivo listo para ser enviado al nanosatélite.

4.3. DISEÑO E IMPLEMENTACIÓN DEL SOFTWARE.

4.3.1. DISEÑO.

4.3.1.1. Diseño de datos.

En el diseño de la base de datos del software, y luego de haber analizado los datos que maneja el software, se concluye que contendrá las mediciones de los subsistemas, y los identificativos de los usuarios del software.

Es así que, se identifican las mediciones principales y son: Voltaje y corriente, Temperaturas Interiores, Generadores de Energía Eléctrica, posicionamiento del nanosatélite y opcionalmente Temperaturas exteriores.

Cada uno de estos, que pueden ser llamados entidades,
tienen sus propios atributos.



Estructuras de archivo y bases de datos.

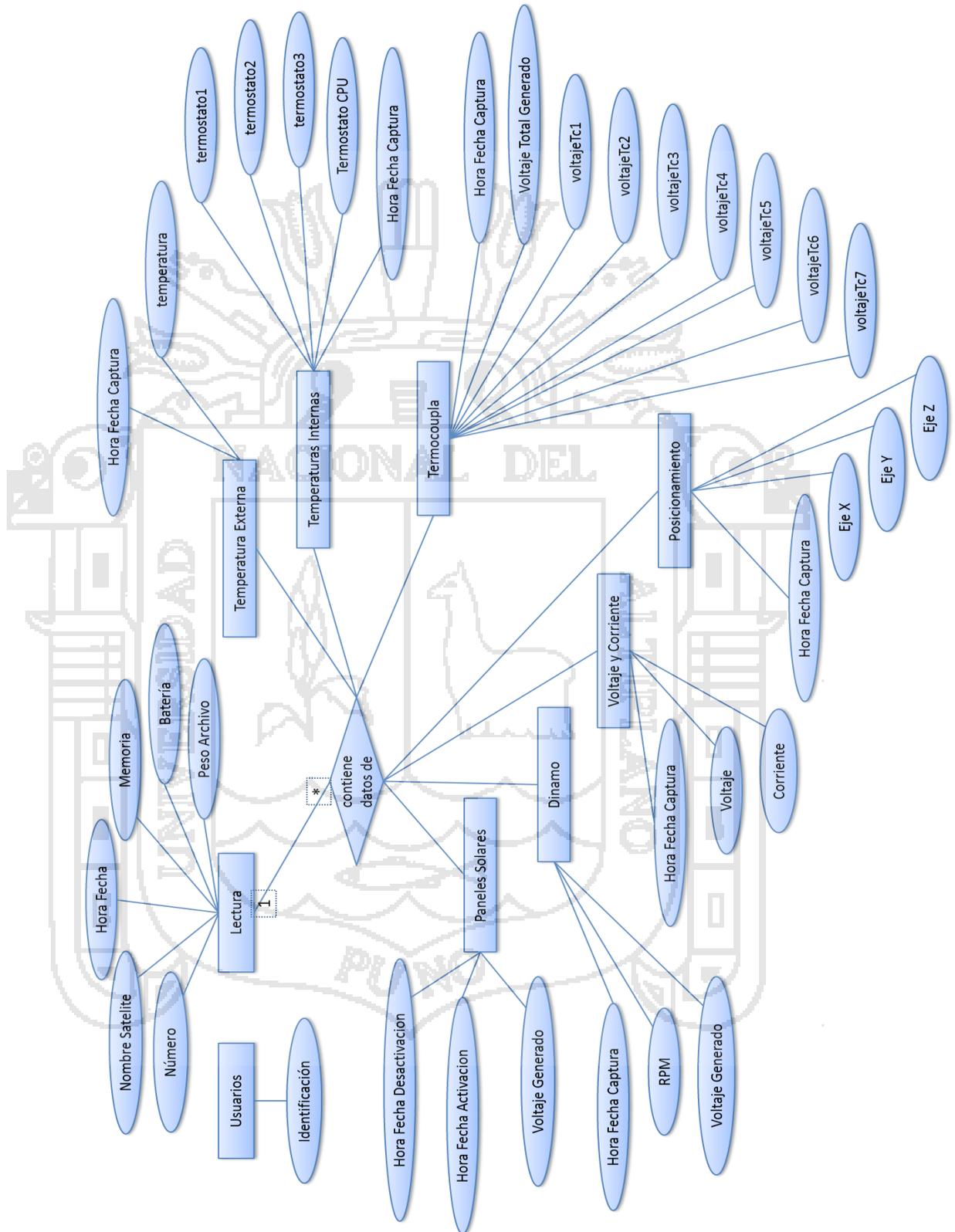


Figura N° 7. Diagrama entidad – relación. Fuente: elaboración propia.

Como se muestra en el diagrama Entidad – Relación, las entidades son: Usuarios, Lectura, Paneles solares, Dinamo, Voltaje y Corriente, Posicionamiento, Temperatura Externa, Temperaturas Internas y Termocoupla. Usuarios no mantiene relación alguna con las demás entidades y la única relación existente es que una lectura contiene varios datos del resto de entidades.

Estructura Lógica.

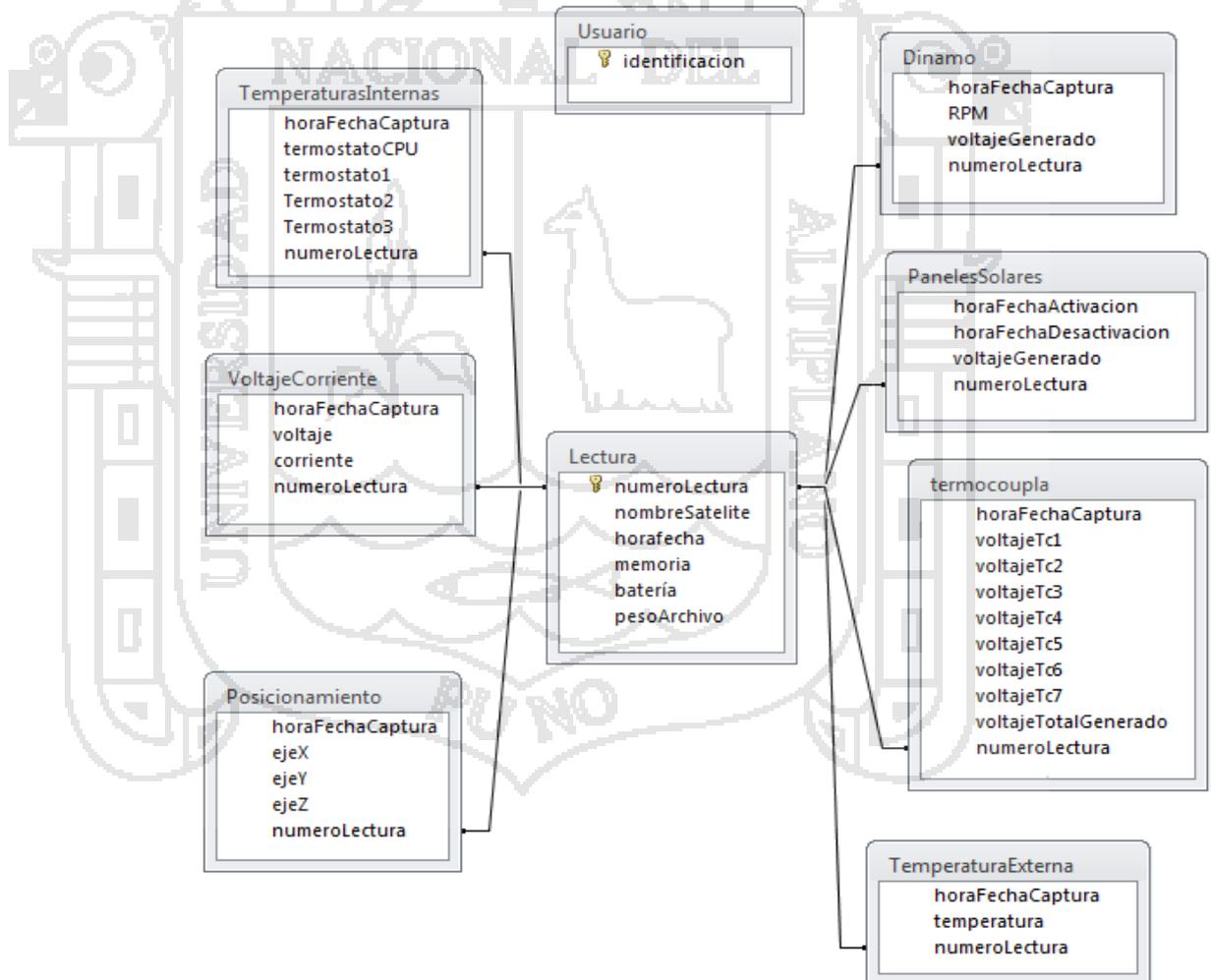


Figura N° 8. Estructura lógica de la base de datos. Fuente: elaboración propia.

La base de datos contendrá la estructura lógica mostrado en la Figura N° 8, de preferencia en el mismo orden de atributos que se muestra.

Estructura Física.

Como se menciona en uno de los requisitos del software en cuanto a la portabilidad, el software no tendrá un gestor de base de datos externo, el software mismo maneja sus datos. Para lo cual se decidió utilizar archivos de texto plano para guardar la información del software, que permitirá la facilidad de tener el software portable.

Un archivo contendrá la entidad Usuario con la siguiente estructura:

IDENTIFICATIVOS

```
[usuario]  
Cadena_identificativo1  
Cadena_identificativo2  
...  
#
```

Figura N° 9 Estructura física de la base de datos, archivo identificativos. Fuente: elaboración propia.

Se almacenara en un archivo de extensión “.dat” el nombre de la entidad está dentro de corchetes “[]” para diferenciarlo de los valores almacenados, las siguientes líneas serán todos los registros de identificativos. Y para determinar el final de registros, se utiliza el símbolo “#”.

NOMBRE_SATELITE

```

[lecturas]
numeroLectura|nombreSatelite|horaFecha|memoria|bateria|pesoArchivo
#
[panelSolar]
horaFechaActivacion|horaFechaDesactivacion|voltajeGenerado|numeroLectura
#
[dinamo]
horaFechaCaptura|RPM|voltajeGenerado|numeroLectura
#
[termocoupla]
horaFechaCaptura|voltajeTc1|voltajeTc2|voltajeTc3|voltajeTc4|voltajeTc5|voltajeTc6|
voltajeTc7|voltajeTotalGenerado|numeroLectura
#
[temperaturaInterna]
horaFechaCaptura|termostatoCPU|termostato1|termostato2|termostato3|numeroLectura
#
[temperaturaExterna]
horaFechaCaptura|temperatura|numeroLectura
#
[posicion]
horaFechaCaptura|ejeX|ejeY|ejeZ|numeroLectura
#
[voltajeCorriente]
horaFechaCaptura|voltaje|corriente|numeroLectura
#

```

Figura N° 10 Estructura física de la base de datos, archivo nombre_satelite. Fuente: elaboración propia.

Habrán archivos que tendrán el nombre del satélite y almacenara todos los datos de las lecturas y las mediciones.

Todas las entidades tienen los atributos mostrados en la Figura N° 8 y en el mismo orden, para delimitarlos se usa un separador de valores dado por el símbolo "|".

4.3.1.2. Diseño arquitectónico.

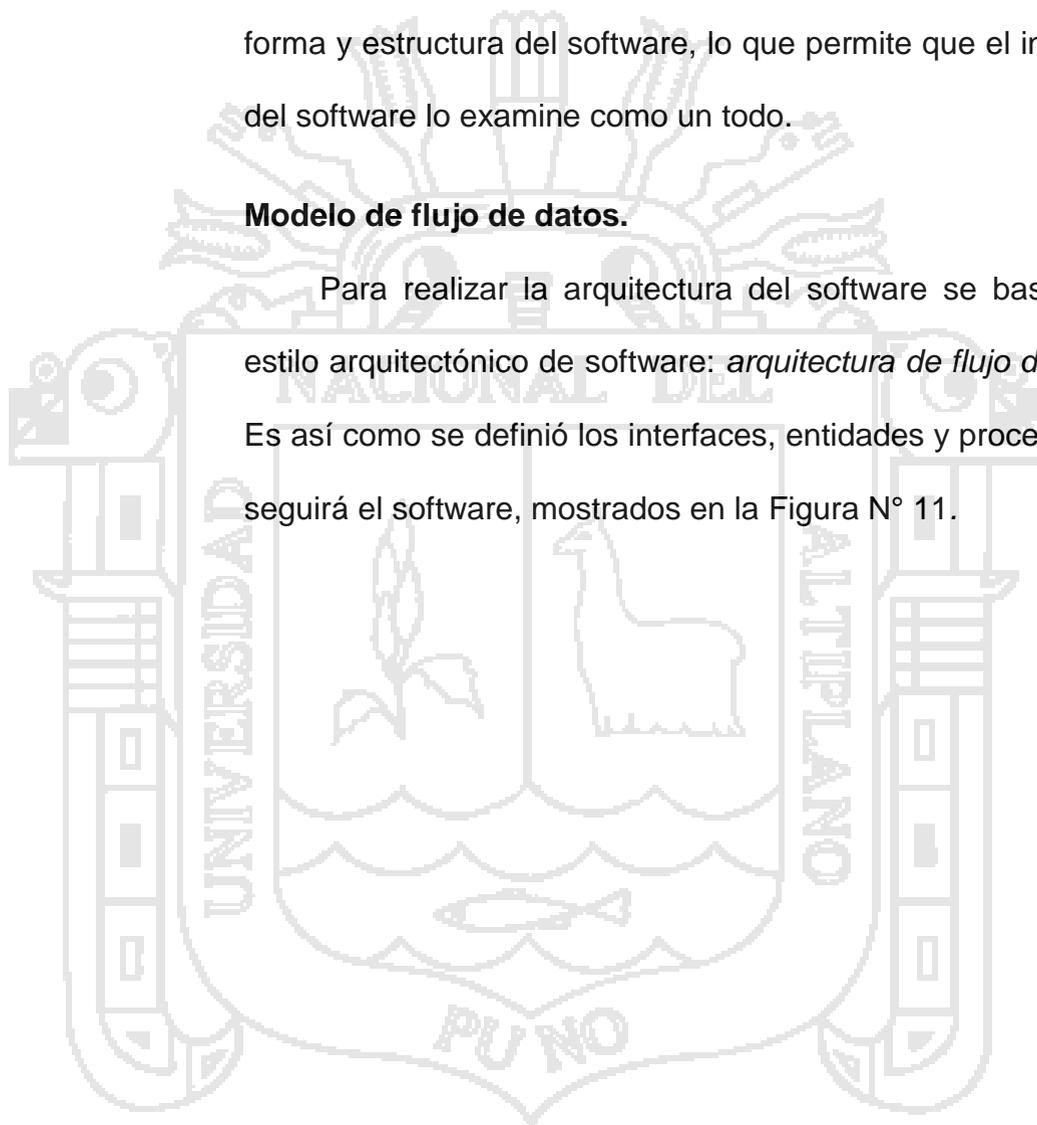
La arquitectura del software es la estructura del programa, lo cual comprende a los componentes del software, sus propiedades externas visibles y las relaciones entre ellos. En el contexto del diseño de la arquitectura, un componente del

software puede ser algo tan simple como un módulo de programa o una clase orientada a objeto (Pressman, 2010).

La arquitectura del software son nuestros planos para la implementación, ya que es un marco general que describe la forma y estructura del software, lo que permite que el ingeniero del software lo examine como un todo.

Modelo de flujo de datos.

Para realizar la arquitectura del software se basó en el estilo arquitectónico de software: *arquitectura de flujo de datos*. Es así como se definió los interfaces, entidades y procesos que seguirá el software, mostrados en la Figura N° 11.



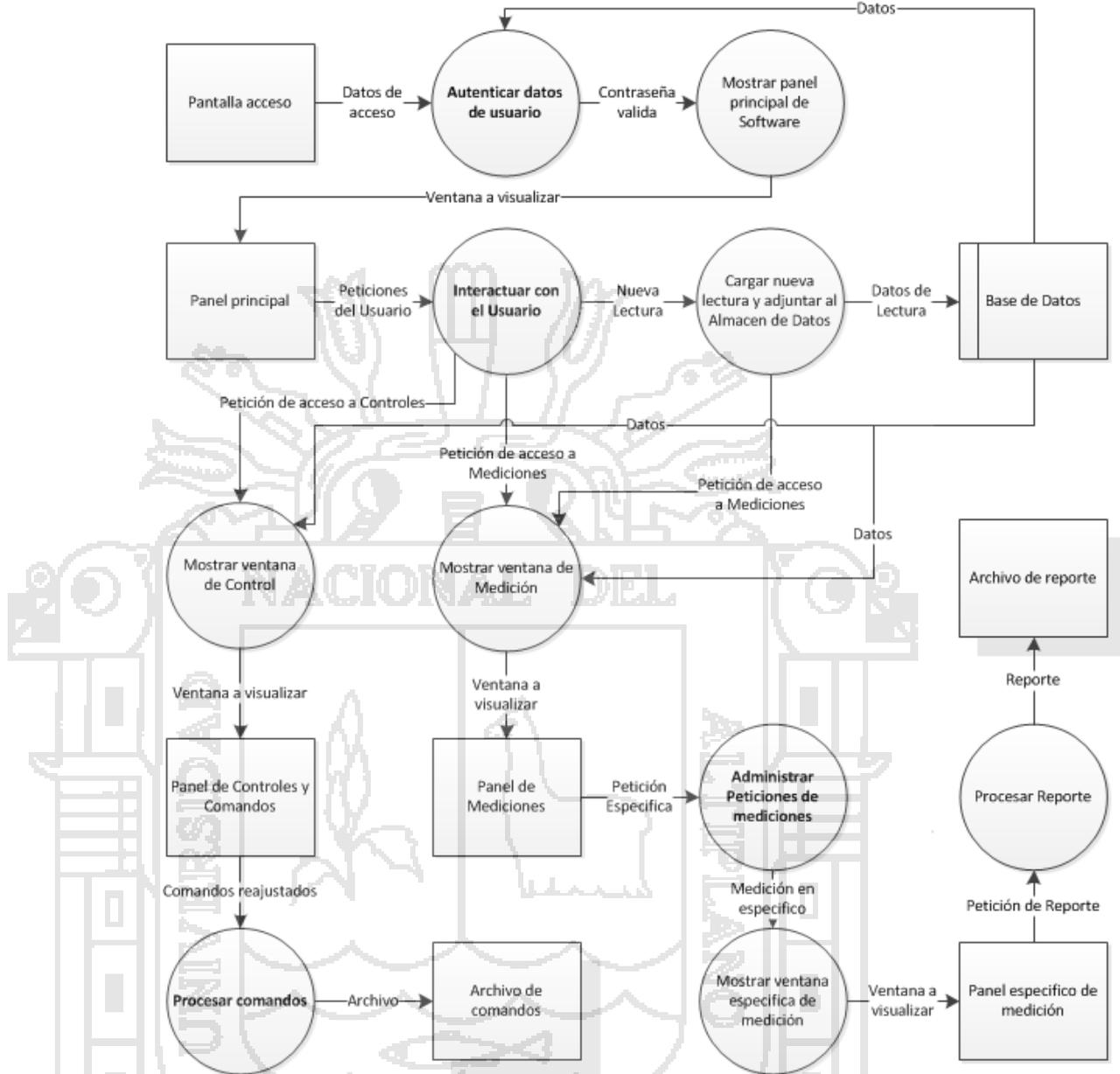


Figura N° 11 Diagrama de flujo de datos del software. Fuente: elaboración propia.

En el diagrama anterior se muestra todos los caminos por el cual fluye la información, se muestra claramente todas las entradas y salidas que se usara en el software, sin embargo es necesario realizar una descomposición de primer nivel, para

refinar los procesos, para este caso se resaltó los procesos que son genéricos.

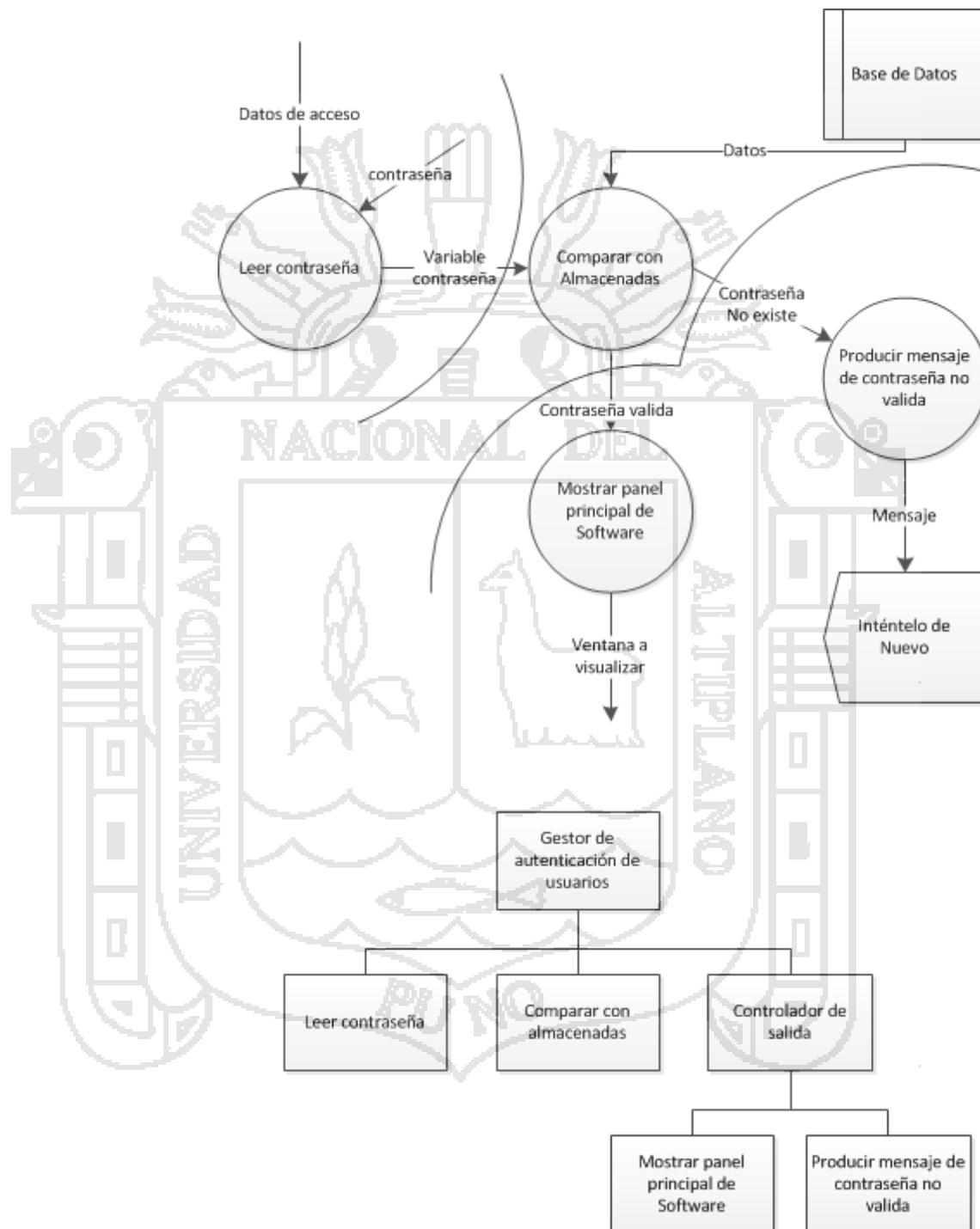


Figura N° 12 Nivel 2 del diagrama de flujo de datos del software, para el proceso Autenticar datos de usuario. Fuente: elaboración propia.

La descomposición mostrada en la Figura N° 12 proporciona una estructura del software, donde se van generando módulos, estos módulos proporcionan control para la trayectoria de flujo de entrada, transformación y de salida. Para ello existirá un controlador principal, para este caso un *Gestor de autenticación de usuarios*, que sirve para coordinar las funciones de control siguientes:

- Controlador de trayectoria de flujo de entrada, denominado Leer contraseña, quien coordina la recepción de los datos de entrada.
- Controlador de trayectoria de flujo de salida, denominado controlador de salida, el cual coordina la producción de la salida, para el caso existen 2 salidas alternativas.
- Y un controlador del flujo de transformación, denominado Comparar con almacenadas, realizara la actividad de comparar la contraseña ingresada por el usuario con las almacenadas en base de datos.

En la Figura N° 13 se muestra el refinamiento del proceso Interactuar con el usuario, en el cual, se observa tres caminos a los que el usuario podrá seguir: acceder a los controles, acceder a mediciones, y cargar una nueva lectura.

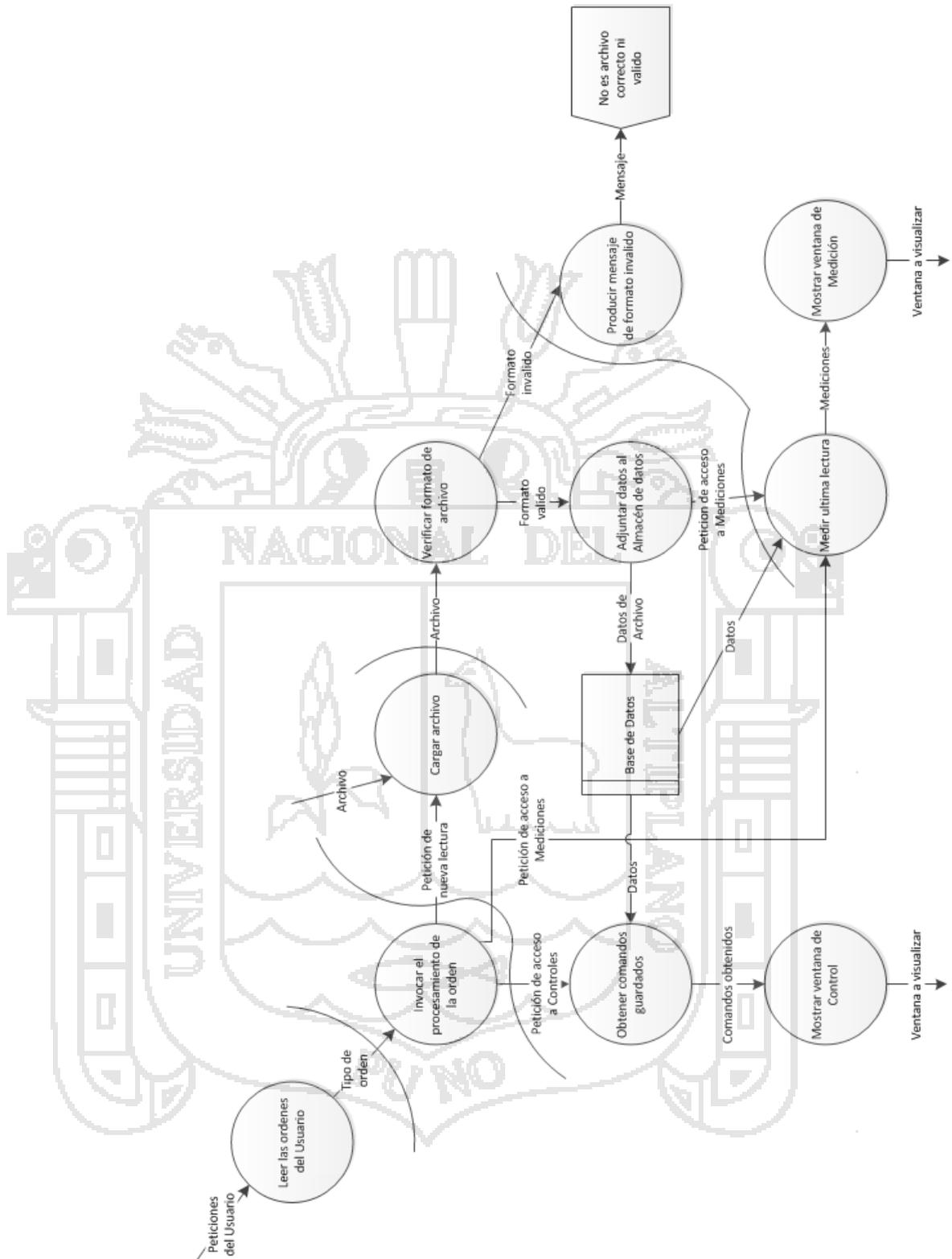


Figura N° 13 Nivel 2 del diagrama de flujo de datos del software, para el proceso Interactuar con el usuario. Fuente: elaboración propia.

De la figura anterior también se obtiene una estructura parcial, la cual se muestra en la Figura N° 14.

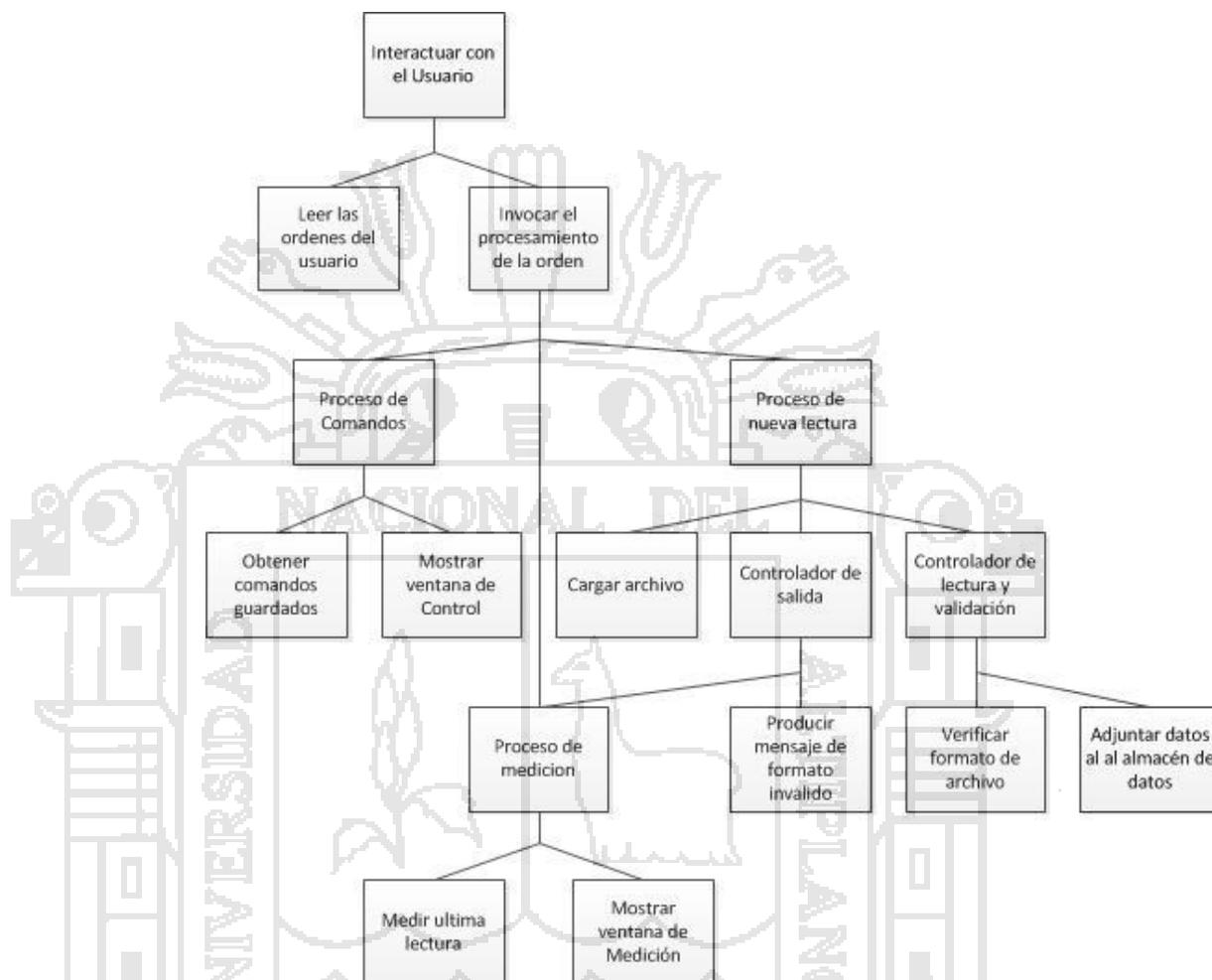


Figura N° 14 Estructura del software para el proceso Interactuar con el usuario. Fuente: elaboración propia.

El proceso Interactuar con el Usuario tendrá un controlador de entrada denominado *Leer las órdenes del usuario*, y otro de procesamiento denominado *Invocar el procesamiento de la orden*. Este último tendrá como subprocesos a: Proceso de comandos, Proceso de medición y Proceso de Nueva lectura. El primero tiene un controlador de

procesamiento denominado *Obtener comandos guardados*, y un controlador de salida nombrado *Mostrar ventana de Control*; lo mismo ocurre con el subproceso Proceso de medición, un controlador de procesamiento *Medir última lectura* y un controlador de salida *Mostrar ventana de Medición*.

En el sub proceso de *Nueva Lectura* se presentan 3 controladores, tanto de entrada, procesamiento y de salida. De controlador de entrada hará únicamente el proceso *Cargar Archivo*, el controlador de procesamiento es el denominado *Controlador de lectura y validación* que tienen los procesos de *Verificar formato de archivo*, y *Adjuntar datos al almacén de datos*; el Controlador de salida llamado así, tendrá un proceso de producir un mensaje de error y otro que llamara al subproceso Proceso de medición.

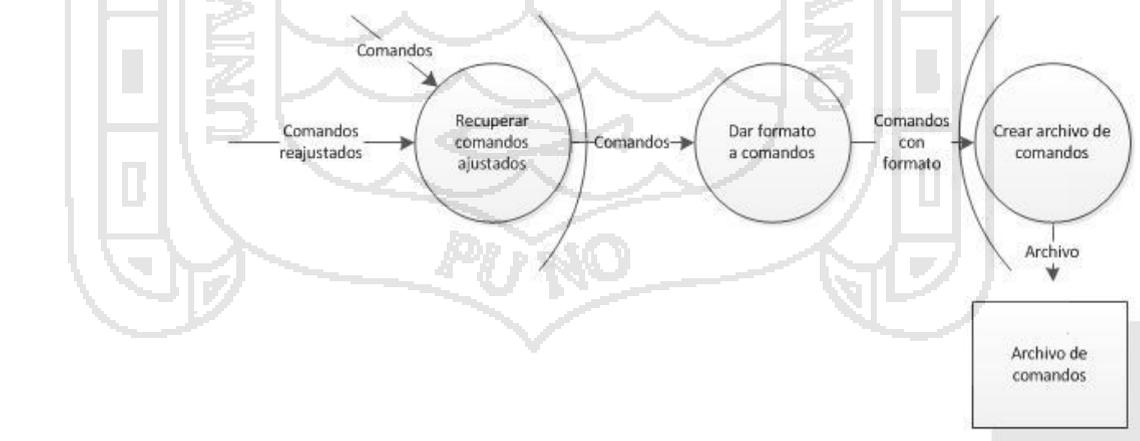


Figura N° 15 Nivel 2 del diagrama de flujo de datos del software, para el proceso Procesar comandos. Fuente: elaboración propia.

El proceso *Procesar comandos*, mostrado en la Figura N° 15, tiene un controlador de entrada llamado *Recuperar*

comandos ajustados, un controlador de procesamiento nombrado *Recuperar comandos ajustados*, y un controlador de salida denominado *Crear archivo de comandos*.

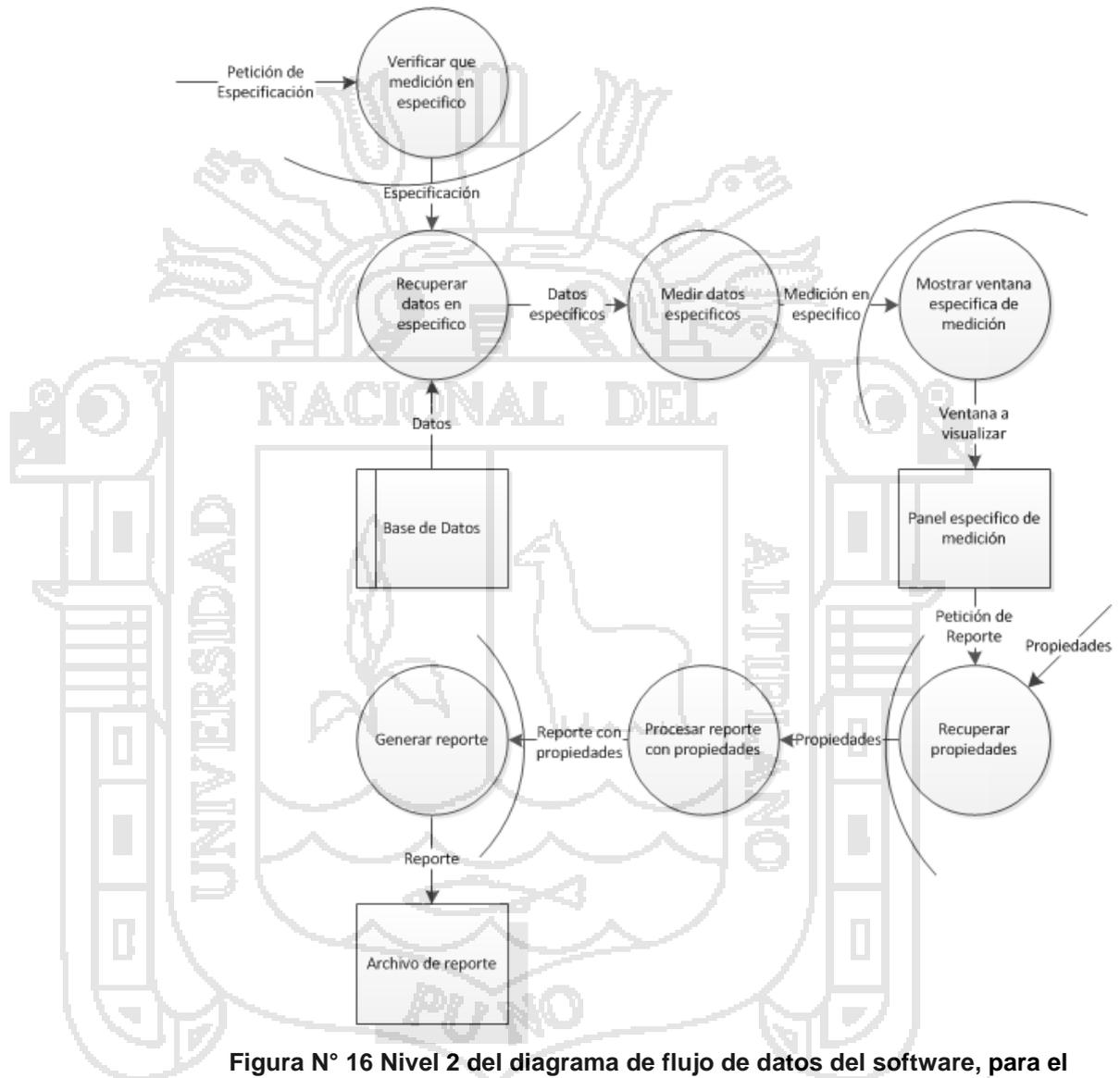


Figura N° 16 Nivel 2 del diagrama de flujo de datos del software, para el proceso Administrar peticiones de mediciones. Fuente: elaboración propia.

Este proceso mostrado en la anterior figura tiene un controlador de entrada llamado *Verificar que medición en específico*, un controlador de transformación, conformado por los

procesos: *Recuperar datos en específico y Medir datos específicos*, y un controlador de salida denominado *Mostrar especifica de medición*; que a partir de él sale otro proceso que puede ser llamado *Generación de Reporte*, que tendrá de igual modo un controlador de entrada *Recuperar propiedades*, un controlador de procesamiento *Procesar reporte con propiedades*, y un controlador de salida *Generar reporte*.

Estructura del programa.

Luego de haber visto y diseñado el diagrama de flujo de datos del software y haber refinado los procesos que eran generales, se obtiene una estructura general del software (ver Figura N° 17), el cual nos guiara durante todo el proceso de desarrollo del software.

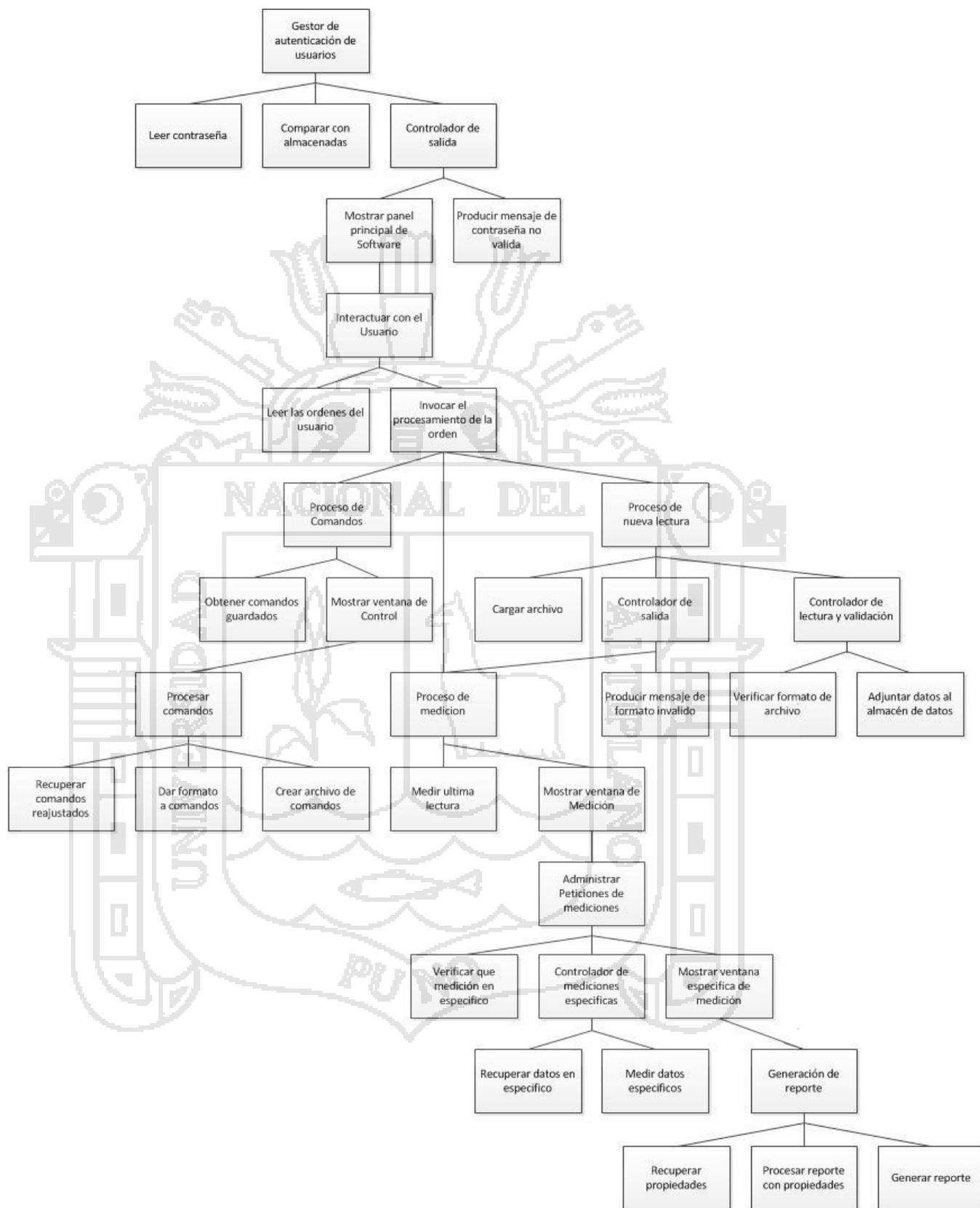


Figura N° 17 Estructura del software. Fuente: elaboración propia.

4.3.1.3. Diseño de interfaz.

Especificación de la interfaz.

El icono diseñado para el software será el siguiente:

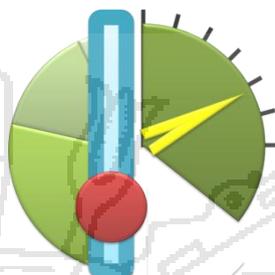


Figura N° 18 Icono del software. Fuente: elaboración propia.

La ventana de apertura del software, será la referente al proceso Autenticar Usuario, donde el usuario tendrá que ingresar su identificación para comenzar a utilizar el software.

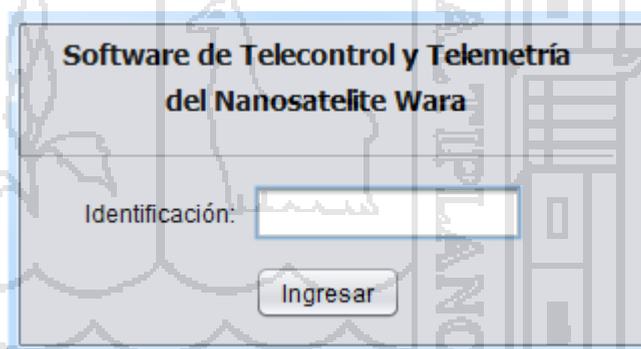


Figura N° 19 Interfaz de acceso al software. Fuente: elaboración propia.

Al ser la primera ventana que se visualizara, se debe mostrar la presentación del software, incluido título y logo identificativo (opcional).



MEDICIÓN	CONTROL
Acceder a mediciones	Acceder a controles y comandos
<input type="button" value="INGRESAR"/>	<input type="button" value="INGRESAR"/>
Cargar nueva lectura	
<input type="text"/> <input type="button" value="Seleccionar"/>	
<input type="button" value="CARGAR E INGRESAR"/>	

Figura N° 20 Interfaz de interacción con el usuario. Fuente: elaboración propia.

En la Figura N° 20 apreciamos el interfaz donde se interactúa con el usuario para las funciones principales del software, en cuanto a la medición y control. Se muestra claramente que la función nueva lectura está dentro de medición, puesto como se analizó en el apartado anterior ambos coinciden en abrir la ventana de mediciones.

Para realizar una nueva lectura, se tendrá una ventana de dialogo con opción de selección de archivos (ver Figura N° 21).

La ventana de telemetría es donde se muestra toda la medición del nanosatélite en general, con opciones para ver más detalle y/o su historial (ver Figura N° 22).

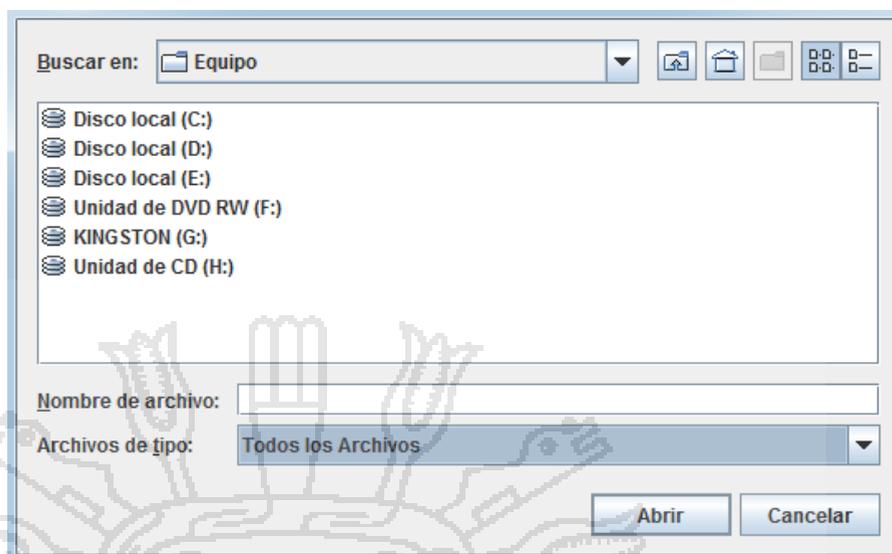


Figura N° 21 Interfaz de selector de archivos. Fuente: elaboración propia.



Figura N° 22 Interfaz de medición. Fuente: elaboración propia.

Las ventanas de detalle e historial de la Figura N° 23 grafican todas las lecturas cargadas al software, en cuadros con gráficos lineales con respecto a la fecha y hora, y ofrecen las posibilidades de realizar reportes.

reguladores, para programar comandos que requieran ser enviados.

4.3.2. IMPLEMENTACIÓN.

4.3.2.1. Para el usuario.

Requerimientos mínimos del software.

Para la utilización del software será necesaria la instalación de la plataforma Java en cualquiera de sus versiones dependiendo del Sistema operativo que utilice.

Se puede descargar desde su página web directamente.

<http://www.java.com>

Instalación.

Basta con ejecutar y seguir los pasos correspondientes para la instalación del software, el cual creará todos los archivos necesarios para el funcionamiento del software.



Figura N° 25 Instalador del software de telemetría y telecontrol. Fuente: elaboración propia.

Uso del software.

El software empieza con una pantalla de Acceso donde se ingresará la contraseña predeterminada.



Figura N° 26 Pantalla de acceso. Fuente: elaboración propia.

Una vez verificado se visualiza la pantalla principal menú donde muestra las opciones de acceder tanto a Mediciones y Controles.

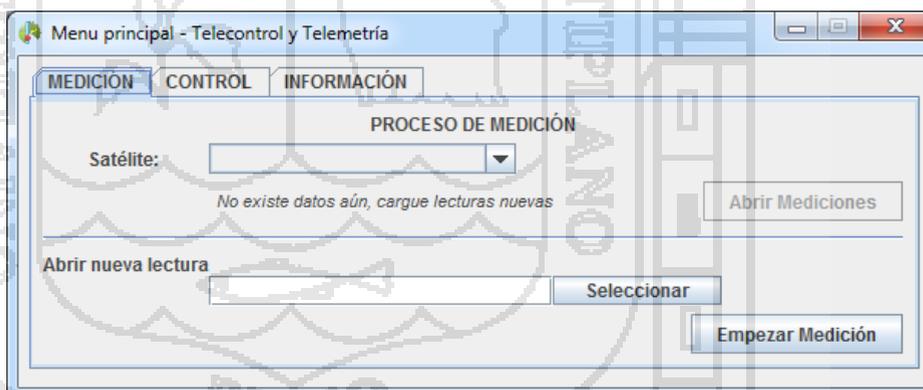


Figura N° 27 Pantalla principal de menú. Fuente: elaboración propia.

Dentro de las opciones de Medición se puede cargar las nuevas lecturas que se reciben del nanosatélite, seleccionando el archivo y empezara la medición o en su defecto directamente acceder a las mediciones con el botón “Abrir Mediciones”.

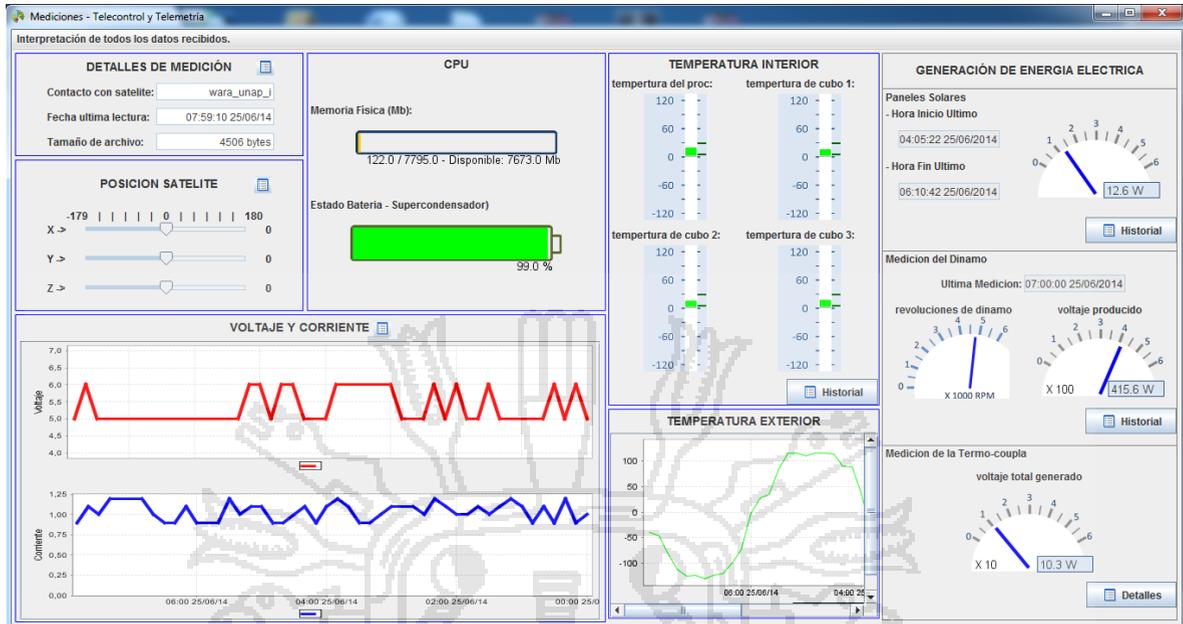


Figura N° 28 Pantalla de medición. Fuente: elaboración propia.

Para los cuadros resumen o historial de todas las medidas hechas, existen botones que abrirán las pantallas y se visualizara al detalle de algunos sensores en específico.

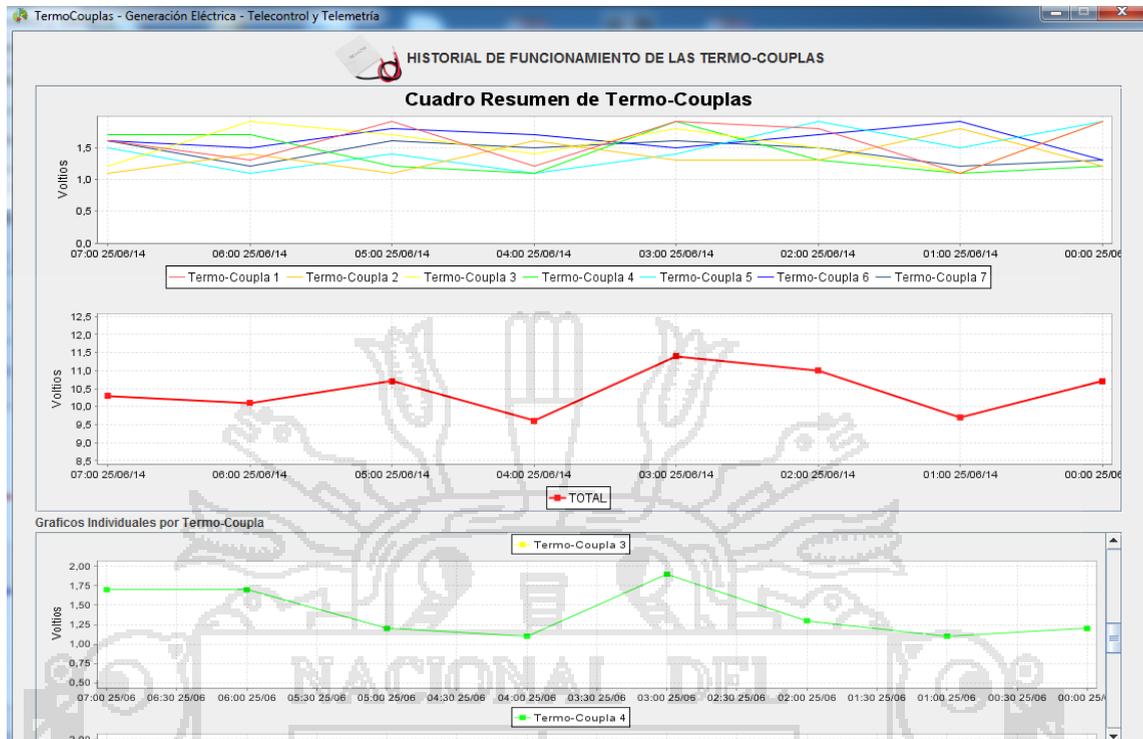


Figura N° 29 Pantalla de medición en específico. Fuente: elaboración propia.

Del mismo modo la pantalla de Control, donde se muestran todos los controles para poder regular, programar y demás funciones para mandarle al nanosatélite.

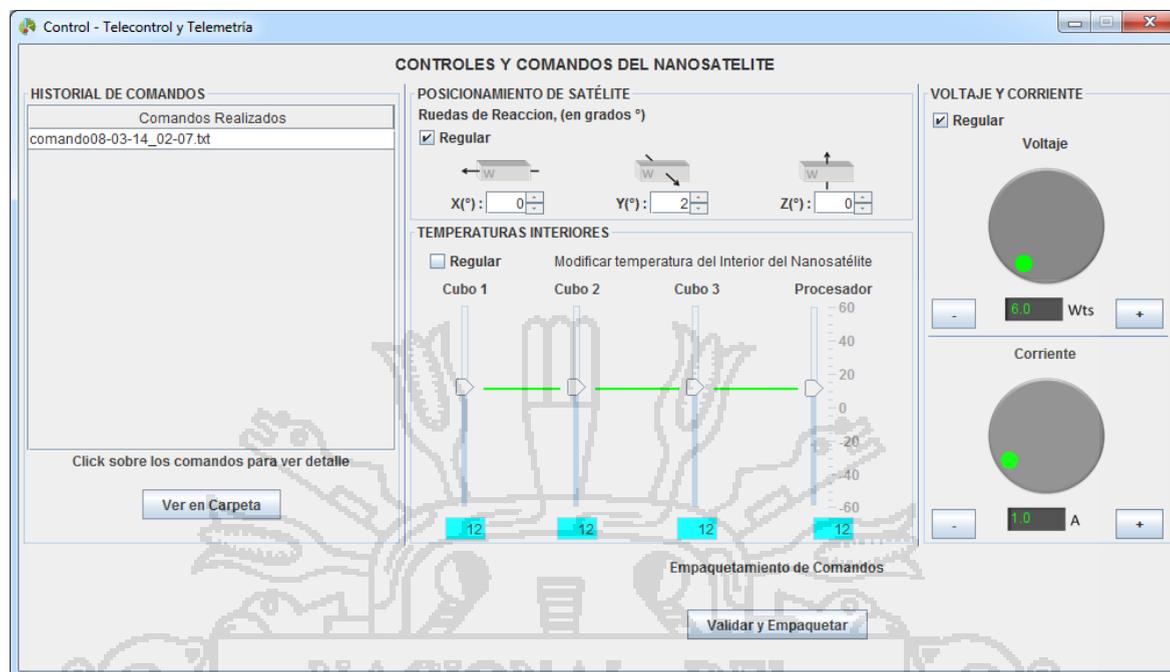


Figura N° 30 Pantalla de controles. Fuente: elaboración propia.

4.3.2.2. Para el programador.

Lenguaje de programación.

El software fue implementado íntegramente en el lenguaje de programación JAVA, y la librería más importante utilizada fue JFreeChart, la cual nos permite crear distintos diagramas estadísticos, puesto q la mayoría de medidas históricas hechas fueron mostradas en gráficos en base al tiempo.

Base de datos.

El software gestiona los datos que fluyen por el mismo, por lo cual no se utiliza ningún gestor de Base de Datos, entonces se almacenan los datos en archivos planos, para el caso se crearon archivos de extensión .DAT.

Los archivos están almacenados en la carpeta de ubicación del software, dentro de la carpeta “data”. Si considera necesario la visualización de los datos a nivel físico, abra con un editor de texto y podrá ver los datos. Mas se advierte tenga **precaución** si se modifican alguno de los datos, lo cual no se recomienda.

Tenga prioridad de crear copias de seguridad de la carpeta q almacena los datos, para así reemplazar la carpeta si es que necesitara reinstalar el software o compartir en otras computadoras.

4.4. EVALUACIÓN DEL SOFTWARE EN EL PROCESO DE TELEMETRÍA Y TELECONTROL.

Para evaluar el software de Telecontrol y Telemetría, se simularon los datos crudos, es decir se crearon archivos lectura con datos crudos tal y como nos enviaría el nanosatélite. Luego se cargaron estos archivos lecturas al software para que pueda realizar las mediciones correspondientes.

Se aplicó un cuestionario Pre prueba para evaluar el contexto sin el software para el proceso de medición y control. Luego se aplicó el tratamiento (Uso del software), y se aplicó después la Post prueba.

El cuestionario planteado fue sometido a la prueba de confiabilidad, utilizando el coeficiente alfa de Cronbach, debemos tener en cuenta que si

obtenemos 0.25 en el coeficiente, esto indica baja confiabilidad; si el resultado es 0.50, la fiabilidad es media o regular. En cambio, si supera el 0.75 es aceptable, y si es mayor a 0.90 es elevada.

N°	Pregunta							Promedio
	1	2	3	4	5	6	7	
1	1	1	1	1	3	1	1	1.14
2	2	1	2	2	1	2	2	1.71
3	1	1	1	2	1	1	1	1.14
4	2	2	1	2	1	1	2	1.57
5	2	2	2	2	1	1	2	1.71
6	2	2	2	2	2	2	3	2.14
7	2	2	2	2	2	2	2	2.00
8	1	1	1	1	1	1	1	1.00

Tabla N° 18. Tabla de consistencia de resultados de pre-prueba. Fuente: elaboración propia.

Analizando la fiabilidad con el coeficiente alfa de Cronbach, mediante el software estadístico:

		N	%
Casos	Validos	8	100,0
	Excluidos	0	,0
	Total	8	100,0

Tabla N° 19. Casos procesados. Fuente: elaboración propia.

Alfa de Cronbach	N de Ítems
,886	7

Tabla N° 20. Estadístico de fiabilidad. Fuente: elaboración propia.

El valor del coeficiente oscila entre: $0.75 < 0.886 < 1.0$, por lo que se considera que la confiabilidad del cuestionario es aceptable.

De igual modo, el cuestionario fue debidamente validado aplicando el juicio por expertos, la validación del instrumento (cuestionario) puede ser revisado en el Anexo B del presente trabajo de investigación.

Los resultados de las medias obtenidos de la Pre prueba y la Post prueba se muestra en la Tabla N° 21.

N°	Pre prueba Sin Software (SS)	Post prueba Con Software (CS)
1	1.14	4.29
2	1.71	4.57
3	1.14	4.14
4	1.57	3.57
5	1.71	4.29
6	2.14	4.43
7	2.00	4.00
8	1.00	3.00

Tabla N° 21. Matriz de datos. Fuente: elaboración propia.

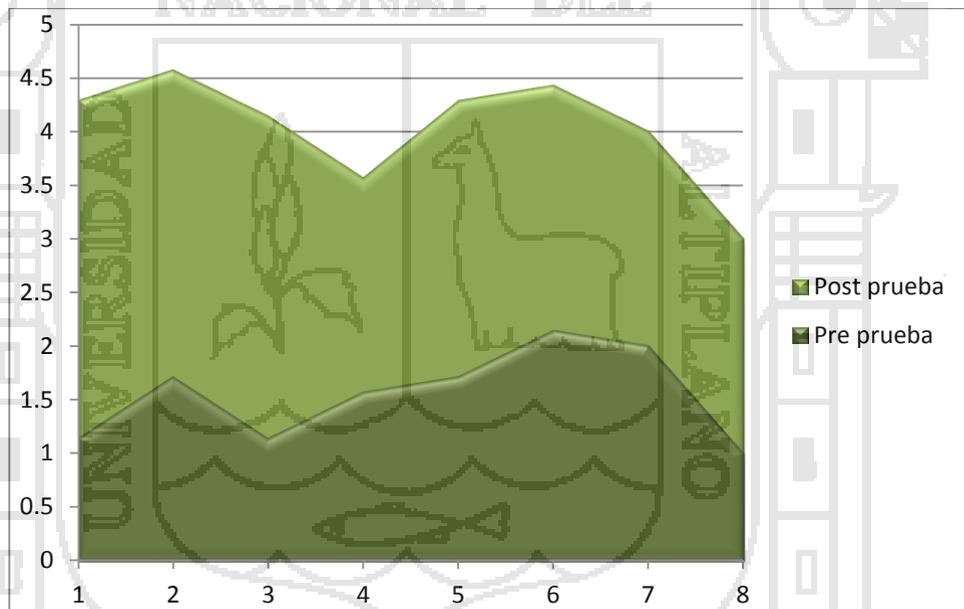


Figura N° 31. Gráfico de matriz de datos. Fuente: elaboración propia.

4.4.1. PRUEBA DE HIPÓTESIS.

H_0 : El uso del software en la Estación Terrena, NO optimiza los procesos de Telemetría y Telecontrol del nanosatélite Wara.

H₁: El uso del software en la Estación Terrena, optimiza los procesos de Telemetría y Telecontrol del nanosatélite Wara.

Aplicaremos la prueba *t* de Student para muestras relacionadas para encontrar una diferencia significativa entre la pre prueba y la post prueba.

Se consideró un nivel de significancia de 0.05.

		Media	N	Desviación típ.	Error típ. de la media
Par 1	PostPrueba	4,04	8	,516	,183
	PrePrueba	1,553	8	,42128	,14895

Tabla N° 22. Estadísticos de muestras relacionadas. Fuente: elaboración propia.

La correlación de las muestras se visualiza en la Tabla N° 23.

		N	Correlación	Sig.
Par 1	PostPrueba y PrePrueba	8	,499	,208

Tabla N° 23. Correlaciones de muestras relacionadas. Fuente: elaboración propia.

Finalmente la prueba de muestras relacionadas.

	Diferencias relacionadas					T	GI	Sig. bilateral
	Media	Desviación típ.	Error típ. de la media	95% Intervalo de confianza para la diferencia				
				Inferior	Superior			
PostPrueba - PrePrueba	2,4821	,47649	,16846	2,0837	2,8805	14,734	7	,000

Tabla N° 24. Prueba de muestras relacionadas. Fuente: elaboración propia.

4.4.2. DECISIÓN.

Podemos comprobar que en los resultados de nuestro caso sí hay diferencias significativas puesto que la significación bilateral es menor que el error permitido del 5% ($0.00 < 0.05$), incluso mucho menor que 1%. Y la diferencia de proporciones está a favor de la post prueba. Por lo tanto se rechaza la Hipótesis nula y se acepta la Hipótesis planteada para la Investigación, El uso del software en la Estación Terrena, optimiza los procesos de Telemetría y Telecontrol del nanosatélite Wara.



CONCLUSIONES

Primero. Se realizó la especificación del software en el que se estableció la descripción general del software, donde una restricción principal es que el software tiene que ser multiplataforma. También especificamos los requerimientos tanto de usuarios, software y de sistema, el requisito funcional que destacamos es la Interpretación de datos crudos obtenido del Nanosatélite, que fue de prioridad muy alta, porque con ello recién se podría: medir y graficar voltaje y corriente, temperaturas interiores, generadores de energía eléctrica y el posicionamiento. Y dentro del proceso de telecontrol: regular voltaje y corriente, temperatura interior; y posicionamiento.

Segundo. Los archivos que contienen los datos crudos tienen una estructura compleja, demorosa y no fiable en la interpretación, entonces en este punto se analizó fila a fila todo el datagrama para lograr la interpretación y poder obtener los datos reales que nos envía el nanosatélite.

Del mismo modo, pero en sentido inverso, el software nos debe permitir realizar órdenes y comandos para ser enviados al nanosatélite y este los ejecute. Dichas órdenes y comandos deben de empaquetarse en forma de dato crudo, para ello también se analizó y diseño el datagrama y su empaquetado que será un archivo de salida.

Tercero. Gracias a la especificación del software y el análisis y diseño de la interpretación de datos crudos y programación de controles, se realizó el

diseño del Software, donde se estableció que el software no tendrá un gestor de base de datos externo, por consiguiente el software mismo manejará sus datos, para ello se utilizará archivos de texto plano para guardar la información del software, estos archivos tendrán la extensión “.dat”. La arquitectura del software fue diseñada mediante un modelo de flujo de datos. Es así que se definió los interfaces, entidades y procesos.

Cumplido el diseño, se pasó a la implementación del software como fue especificado, se utilizó un lenguaje multiplataforma, por ello se utilizó JAVA. Un requerimiento para la utilidad del software es instalar la plataforma Java en la versión dependiendo del Sistema operativo del equipo de cómputo. El software tiene un instalador que contiene todos los archivos necesarios para el funcionamiento del software. Se realizó el informe correspondiente tanto para el usuario final y para el programador.

Cuarto. Se acepta la Hipótesis planteada en el Proyecto de Investigación, y se rechazó la nula, gracias a la prueba de T de student. Así, es como el uso del software en la Estación Terrena, optimiza los procesos de Telemetría y Telecontrol del nanosatélite Wara.

Y con el software aceptado, estamos listos para continuar con el desarrollo de nuestro primer satélite puneño de la mano de nuestra Facultad ingenieril.

RECOMENDACIONES

Primero. En algunos puntos de la especificación del software es necesario establecerlo conjuntamente con la programación del microcontrolador del nanosatélite, específicamente en los requisitos del software; esto para establecer el mismo nombre y tipo de variables de los distintos módulos que tendrá el satélite.

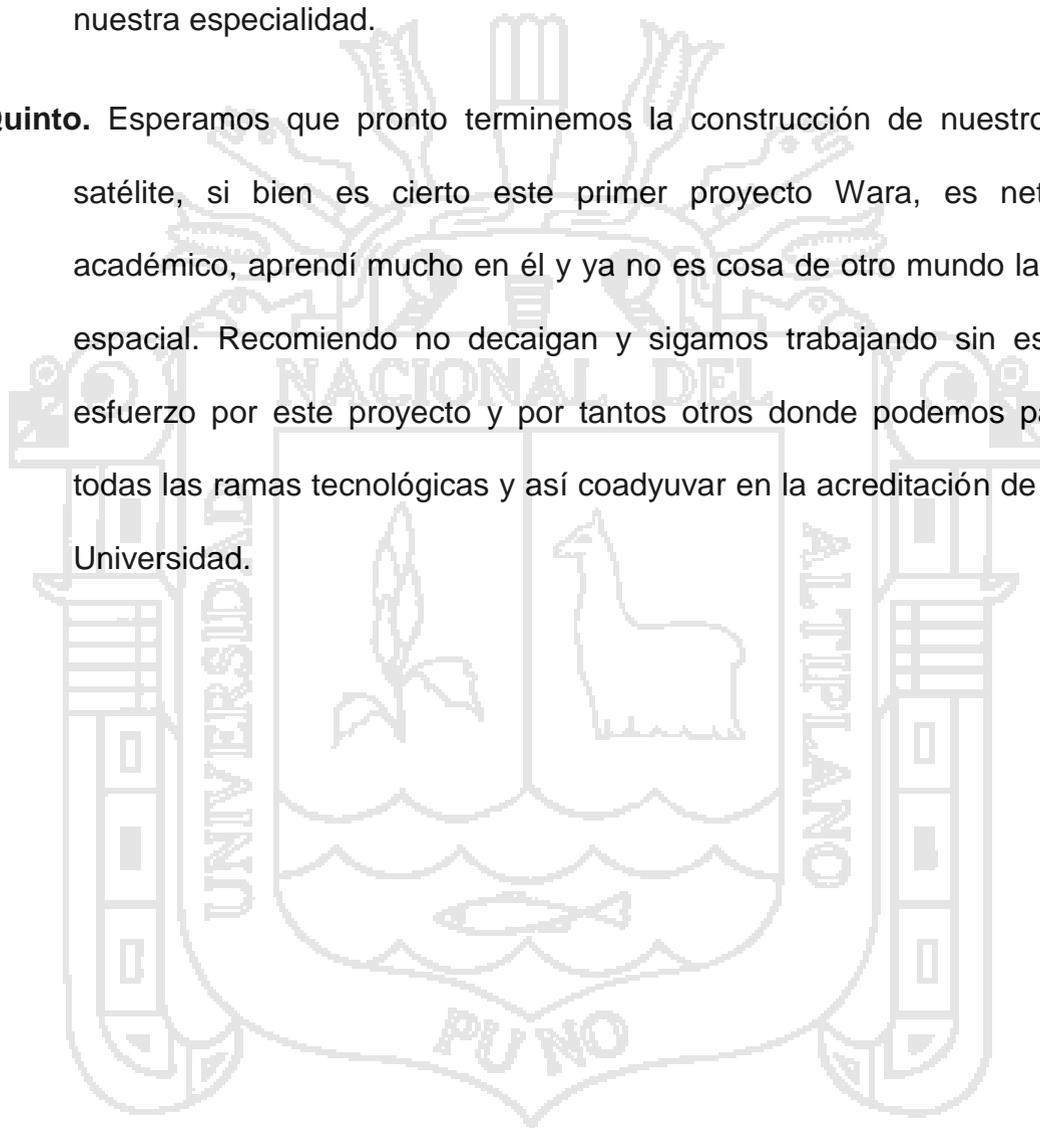
Segundo. La interpretación de los datos crudos y el almacenamiento del mismo fueron almacenados en archivos de texto plano, estos pueden optimizarse utilizando otro tipo de estructura, que permita la facilidad de gestionarlos después en forma estándar con respecto a otros software de Telecontrol y Telemetría existentes en el mundo.

Tercero. Se recomienda continuar con este tipo de investigación, nos referimos al desarrollo de la Tecnología Aeroespacial, como se describió antes, el desarrollo aeroespacial para un país representa su nivel tecnológico, su capacidad de desarrollo y hasta su poderío, el desarrollo de la industria aeroespaciales es de mucho uso en nuestros días, tanto en telecomunicaciones mundial, sistemas de posicionamiento global y marítimo, y falta líneas para describir todos los usos que se le puede dar a las imágenes satelitales.

Cuarto. Sugiero que los próximos graduandos sigan con la exploración de nuevas tecnologías, interdisciplinarias, no solo centrarnos en hacer aplicaciones, software, sistemas específicos; debemos trabajar en equipo, para lograr

proyectos aún más grandes, de mayor trascendencia. Si reflexionamos nuestro entorno, somos la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas, me atrevo a decir la única facultad que vive y desarrolla tecnología, daríamos vida a cuantos proyectos inherentes a nuestra especialidad.

Quinto. Esperamos que pronto terminemos la construcción de nuestro propio satélite, si bien es cierto este primer proyecto Wara, es netamente académico, aprendí mucho en él y ya no es cosa de otro mundo la carrera espacial. Recomiendo no decaigan y sigamos trabajando sin escatimar esfuerzo por este proyecto y por tantos otros donde podemos participar todas las ramas tecnológicas y así coadyuvar en la acreditación de nuestra Universidad.



BIBLIOGRAFÍA

- Canarysat. (14 de Mayo de 2011). *CubeSat*. Recuperado el 21 de Diciembre de 2013, de Canarysat: <http://www.canarysat.com/cubesat.html>
- Computer Hope. (2014). *Computer Hope*. Recuperado el 17 de 06 de 2014, de Computer Hope: <http://www.computerhope.com/jargon/r/rawdata.htm>
- CubeSat. (2013). *Mission Statement*. Recuperado el 20 de Diciembre de 2013, de CubeSat: <http://www.cubesat.org/index.php/about-us/mission-statement>
- Deutsche Welle. (27 de 05 de 2014). *DW Deutsche Welle*. Recuperado el 27 de 06 de 2014, de <http://www.dw.de/telemetr%C3%ADa-el-lenguaje-de-los-sat%C3%A9lites/a-17665651>
- European Space Agency. (01 de 05 de 2014). *ESA*. Recuperado el 27 de 06 de 2014, de ESA:
http://www.esa.int/Our_Activities/Space_Engineering/Onboard_Computer_and_Data_Handling/Telemetry_Telecommand
- García, B., & Llorente, R. (2013). *Clasificación de los Satélites Artificiales*. Recuperado el 21 de Diciembre de 2013, de Laboratorio de Procesado de Imagen:
http://www.lpi.tel.uva.es/~nacho/docencia/EMC/trabajos_02_03/RADIOASTRONOMIA/web/Indice/S_art/I_s_art/3_3/Clas.htm

Grupo Hispasat. (11 de 02 de 2014). *Hispasat*. Recuperado el 18 de 03 de 2014, de: <http://www.hispasat.com/contenidos/web-documentos/af--redes-telecontrol-esp11-02-14.pdf>

Hernández Loja, M. A., & Rojas Advíncula, A. F. (2008). *Modulo de Comunicación Satelital*. Universidad Nacional de Ingeniería, Centro de Tecnologías de Información y Comunicaciones, Lima - Perú.

Instituto Nacional de Tecnologías de la Comunicación. (2009). *Ingeniería del software: Metodologías y ciclos de vida*. España.

Manzano Ramos, E. A. (2013). *Diseño e implementación del software para el cálculo de posición del picosatélite perteneciente al proyecto QB50 utilizando Mecánica Orbital*. Universidad Nacional del Altiplano, Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas, Puno - Perú.

Mozombite Frisancho, A. V. (2012). *Diseño de un Nanosatélite para el Proyecto de Estudio Atmosférico QB50*. Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería, Lima - Perú.

National Aeronautics and Space Administration. (16 de Junio de 2011). *What Is a Satellite?* Recuperado el 20 de Diciembre de 2013, de NASA Education: <http://www.nasa.gov/audience/forstudents/k-4/stories/what-is-a-satellite-k4.html>

Neri Vela, R. (2003). *Comunicaciones por Satélite*. Mexico: International Thomson Editores, S.A.

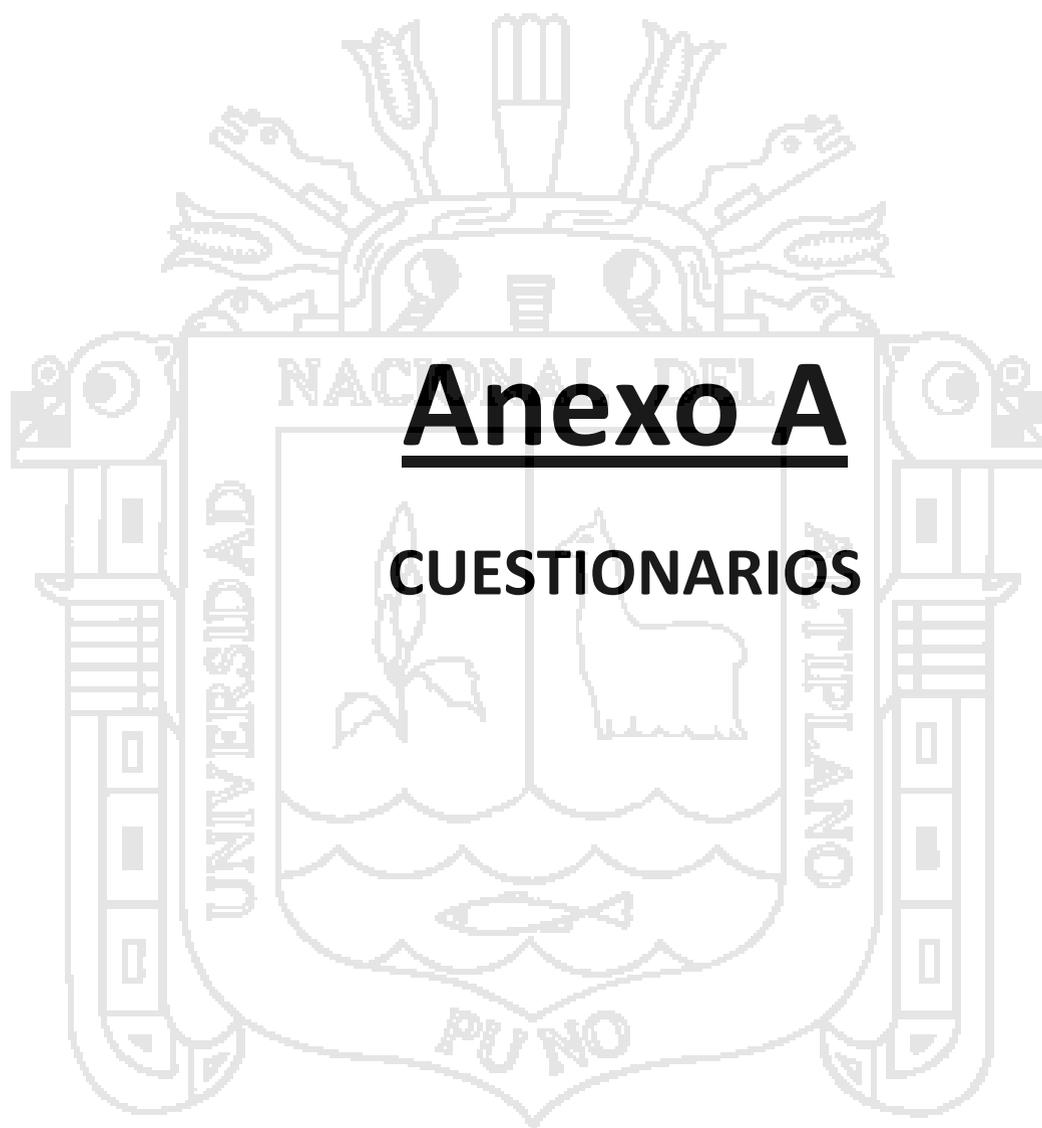
- Peredo, S. (13 de Diciembre de 2004). *Software para analisis del presupuesto de enlace para comunicaciones vía satélite*. Recuperado el 21 de Diciembre de 2013, de UDLAP Bibliotecas:
http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/peredo_a_s/capitulo_3.html
- Pressman, R. S. (2010). *Ingeniería del Software. Un Enfoque Practico. Séptima Edición* (Séptima Edición ed.). Mexico: McGraw-Hill Interamericana Editores, S.A. de C.V.
- Roman Gonzales, A., & Vargas Cuentas, N. I. (2013). *Tecnología Aeroespacial en el Mundo*.
- Rosado, C. (1998). *Comunicación por satélite*. Albadalejo, S.L.
- Sampieri, R. H. (2010). *Metodología de la Investigación, Quinta Edición*. Mexico: McGraw-Hill / Interamericana Editores, S.A. de C.V.
- Sommerville, I. (2005). *Ingeniería del Software, Séptima Edición*. Madrid: Pearson Educación, S.A.
- Verano Galindo, C. A. (2012). *Diseño de un dinamo y Termo-Coupla como alternativas de generación de energía eléctrica para un nanosatélite tipo 3U Cubesat*. Universidad Nacional del Altiplano, Facultad de Ingeniería Mecánica Eléctrica, electrónica y Sistemas, Puno - Perú.

Von Karman Institute and the European Space Agency. (2011). *QB50 an international network of 50 CubeSats*. Recuperado el 2 de Agosto de 2013, de QB50 an international network of 50 CubeSats: www.vki.ac.be/QB50



ANEXOS







UNIVERSIDAD NACIONAL DEL ALTIPLANO – PUNO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS

CUESTIONARIO

Buen día:

Estamos trabajando en un estudio que servirá para elaborar una tesis profesional titulado “Software de Telecontrol y Telemetría para los procesos de Medición y Control del Nanosatélite Wara” estudio que contribuirá al Proyecto Wara de la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas de la UNA Puno.

Quisiéramos pedir su ayuda para que colabore con el estudio, contestando algunas preguntas que no llevara mucho tiempo. Usted es una de las pocas personas entendidas que podría calificar el estudio de la tesis mencionada.

De antemano, Muchas gracias por su colaboración.

Instrucciones:

Marque con un aspa o (X) la casilla que sea su respuesta. Recuerde: NO puede marcar dos opciones por pregunta.

Nombre:

Cargo en el proyecto Wara:

Tipo de Cuestionario: **Pre Test.**

Preguntas sobre el proceso de medición de datos crudos recibidos del nanosatélite.

1. Considera Ud. que el proceso para la medición de los datos crudos recibidos del Nanosatélite es:

Muy sencillo Sencillo Regular Difícil Muy difícil

2. El proceso de medir e interpretar los datos crudos recibidos del Nanosatélite, en cuanto al tiempo es:

Muy rápido Rápido Regular Lento Muy Lento

3. El proceso de medir e interpretar los datos crudos, ¿Es fiable?, es decir, ¿existe el riesgo de equivocarse en la interpretación?

Definitivamente No Probablemente No Indeciso Probablemente si Definitivamente Si

4. En el proceso de medición de los datos crudos, ¿Las mediciones son gráficamente entendibles?

Completamente Bastante Regular Poco Nada

Preguntas sobre el Proceso de Control del Nanosatélite.

5. Considera Ud. que el proceso para programar las órdenes y comandos para el Nanosatélite es:

Muy sencillo Sencillo Regular tedioso Muy tedioso

6. En el proceso de programar los comandos, ¿Se encuentra vulnerable de programar parámetros erróneos?

Definitivamente No Probablemente No Indeciso Probablemente si Definitivamente Si

7. En el proceso de programación de comandos, ¿Los controles son operados gráficamente?

Completamente Bastante Regular Poco Nada





UNIVERSIDAD NACIONAL DEL ALTIPLANO – PUNO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS

CUESTIONARIO

Buen día:

Estamos trabajando en un estudio que servirá para elaborar una tesis profesional titulado “Software de Telecontrol y Telemetría para los procesos de Medición y Control del Nanosatélite Wara” estudio que contribuirá al Proyecto Wara de la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas de la UNA Puno.

Quisiéramos pedir su ayuda para que colabore con el estudio, contestando algunas preguntas que no llevara mucho tiempo. Usted es una de las pocas personas entendidas que podría calificar el estudio de la tesis mencionada.

De antemano, Muchas gracias por su colaboración.

Instrucciones:

Marque con un aspa o (X) la casilla que sea su respuesta. Recuerde: NO puede marcar dos opciones por pregunta.

Nombre:

Cargo en el proyecto Wara:

Tipo de Cuestionario: **Post Test.**

Preguntas sobre el proceso de medición de datos crudos recibidos del nanosatélite.

1. Considera Ud. que el proceso para la medición de los datos crudos recibidos del Nanosatélite, con el uso del software es:

Muy sencillo Sencillo Regular Difícil Muy difícil

2. El proceso de medir e interpretar los datos crudos recibidos del Nanosatélite con el uso del software, en cuanto al tiempo es:

Muy rápido Rápido Regular Lento Muy Lento

3. El proceso de medir e interpretar los datos crudos con el uso del software, ¿Es fiable?, es decir, ¿existe el riesgo de equivocarse en la interpretación?

Definitivamente No Probablemente No Indeciso Probablemente si Definitivamente Si

4. En el proceso de medición de los datos crudos con el uso del software, ¿Las mediciones son gráficamente entendibles?

Completamente Bastante Regular Poco Nada

Preguntas sobre el Proceso de Control del nanosatélite.

5. Considera Ud. que el proceso para programar las órdenes al Nanosatélite, con el uso del software es:

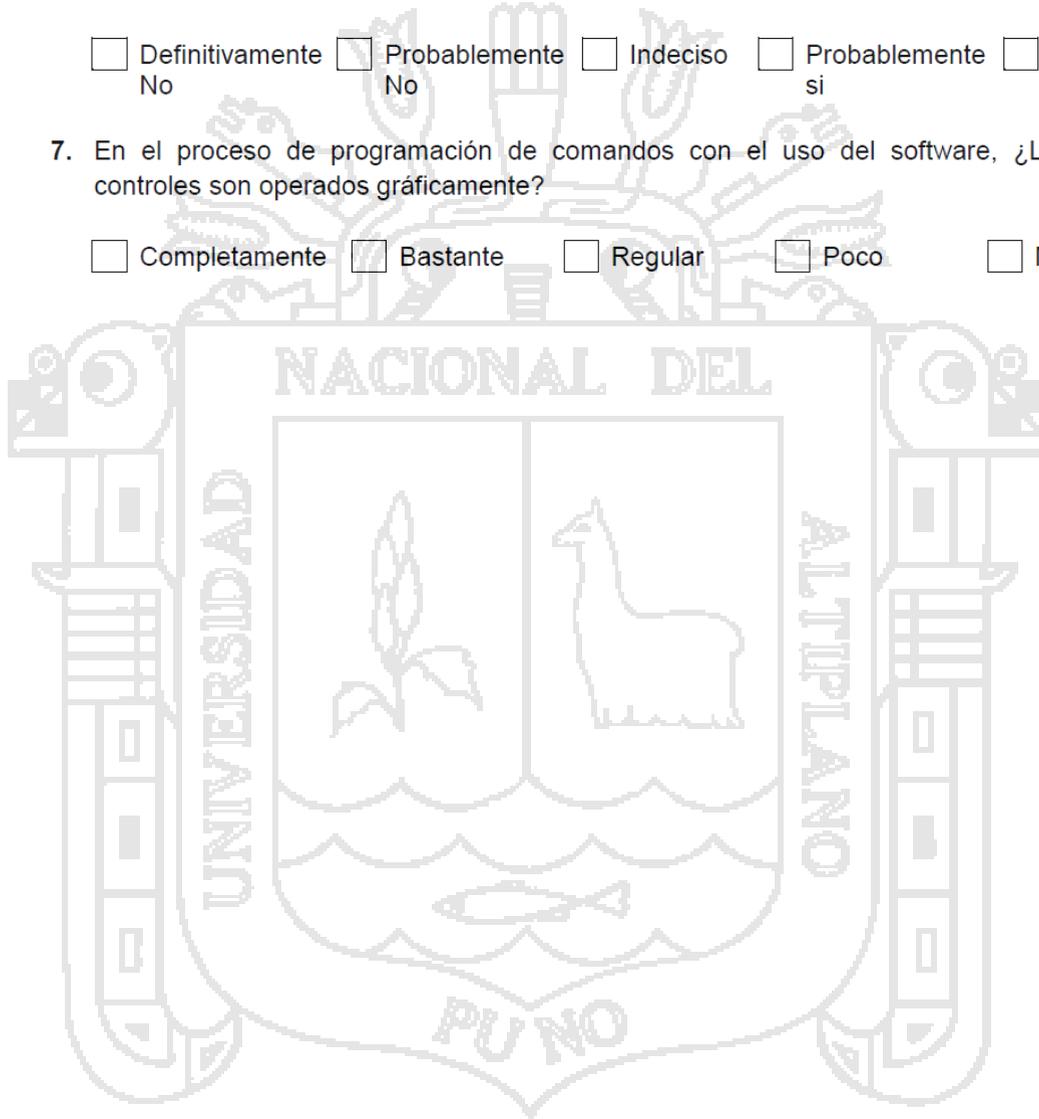
Muy sencillo Sencillo Regular tedioso Muy tedioso

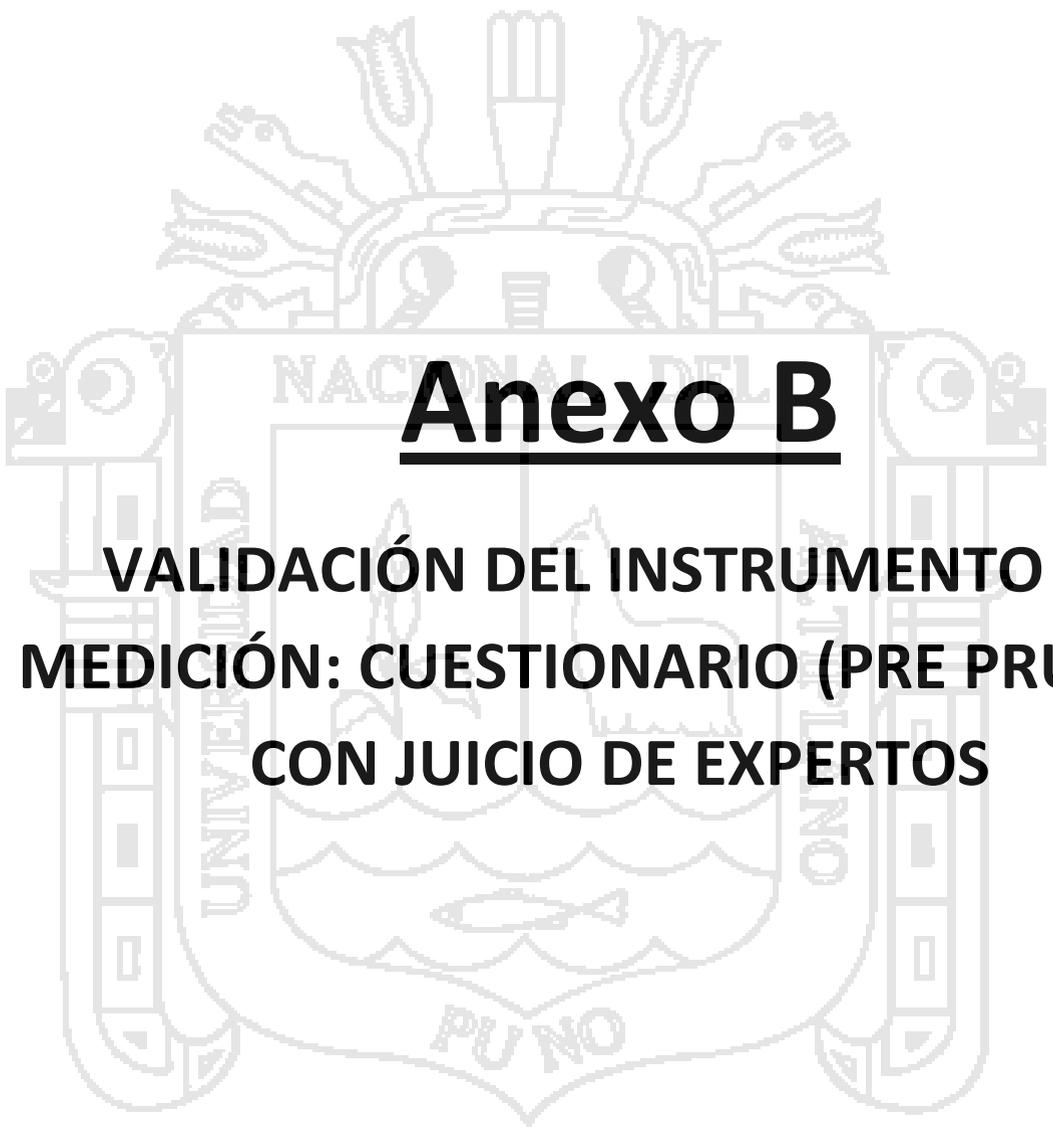
6. En el proceso para programar los comandos con el uso del software, ¿Se encuentra vulnerable de programar parámetros erróneos?

Definitivamente No Probablemente No Indeciso Probablemente si Definitivamente Si

7. En el proceso de programación de comandos con el uso del software, ¿Los controles son operados gráficamente?

Completamente Bastante Regular Poco Nada





Anexo B

VALIDACIÓN DEL INSTRUMENTO DE MEDICIÓN: CUESTIONARIO (PRE PRUEBA) CON JUICIO DE EXPERTOS

SOLICITUD DE VALIDACIÓN DIRIGIDA A EXPERTOS EN:**Telecontrol y Telemetría**

rudol huallpa <rkatarih@gmail.com>

Validación de instrumento a través de juicio de Expertos. WARA-UNAP

rudol huallpa <rkatarih@gmail.com>

2 de agosto de 2014, 3:04

Para: avid.roman-gonzalez@ieee.org, jaime.estela@gmail.com, jmartincanalesr@hotmail.com

Buen día.

Le escribo desde la UNA Puno, espero me recuerde, vine elaborando el Software de Telecontrol y Telemetría para el Nanosatélite Wara. y para lo cual he realizado un instrumento de recopilación de datos (Cuestionario Pre Test), por lo que recorro a Ud. a fin de solicitarle muy cordialmente pueda emitir su juicio hacia el instrumento de investigación a utilizar.

Le agradeceré mucho su pronta respuesta, para el cual adjunto:

- Instrumento que se desea validar (Cuestionario Pre Test).
- Tesis de investigación (los puntos mas importantes son el Capitulo IV: Resultados y Discusión),
- Instalador del software de Telecontrol y Telemetría.

Es propicia la oportunidad para reiterarle las muestras de mi especial consideración y estima personal.

Muy atentamente, Tesista: Rudol Katari Huallpa LLanos.

PD:

le adjunto el formato de Informe de Validación de Expertos, para que pueda dar su juicio en dicho documento, y favor inmenso le pediría, para dar mayor crédito al informe una firma al final y me lo podría escanear y enviármelo por favor. Muchas gracias de antemano.

5 archivos adjuntos

-  **CUESTIONARIO - POST.docx**
39K
-  **CUESTIONARIO - PRE.docx**
46K
-  **INFORME DE VALIDACION DE EXPERTOS.xlsx**
19K
-  **Software TCM.rar**
2287K
-  **TESIS-BORRADOR.pdf**
2755K



ESQUEMA DE INFORME DE VALIDACIÓN DE EXPERTOS

INFORME DE VALIDACIÓN DE EXPERTOS

I. DATOS GENERALES:

- 1.1. Apellidos y Nombres del Informante: _____
- 1.2. Institución donde labora: _____
- 1.3. Título de la Investigación: "Software De Telecontrol Y Telemetría Para Los Procesos De Medición Y Control Del Nanosatélite Wara".
- 1.4. Autor del Instrumento: Rudol Katari Huallpa Llanos.
- 1.5. Nombre del instrumento motivo de evaluación: CUESTIONARIO.

II. ASPECTOS DE VALIDACIÓN:

INDICADORES	CRITERIOS	Deficiente				Baja				Regular				Buena				Excelente			
		0	6	11	16	21	26	31	36	41	46	51	56	61	66	71	76	81	86	91	96
		5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
1. CLARIDAD	Esta formulado con lenguaje apropiado																				
2. OBJETIVIDAD	Esta expresado con conductas observables																				
3. ACTUALIDAD	Adecuado al avance de la ciencia																				
4. ORGANIZACIÓN	Existe una organización lógica																				
5. SUFICIENCIA	Comprende los aspectos en cantidad y calidad																				
6. INTENCIONALIDAD	Adecuado para valorar la gestión tecnológica																				
7. CONSISTENCIA	Basado en aspectos teóricos científicos																				
8. COHERENCIA	Entre los índices e indicadores																				
9. METODOLOGÍA	La estrategia responde al propósito del diagnóstico																				
10. PERTINENCIA	Es útil y adecuado para la investigación.																				

III. OPINIÓN DE APLICABILIDAD:

IV. PROMEDIO DE VALORACIÓN: _____ 0 _____

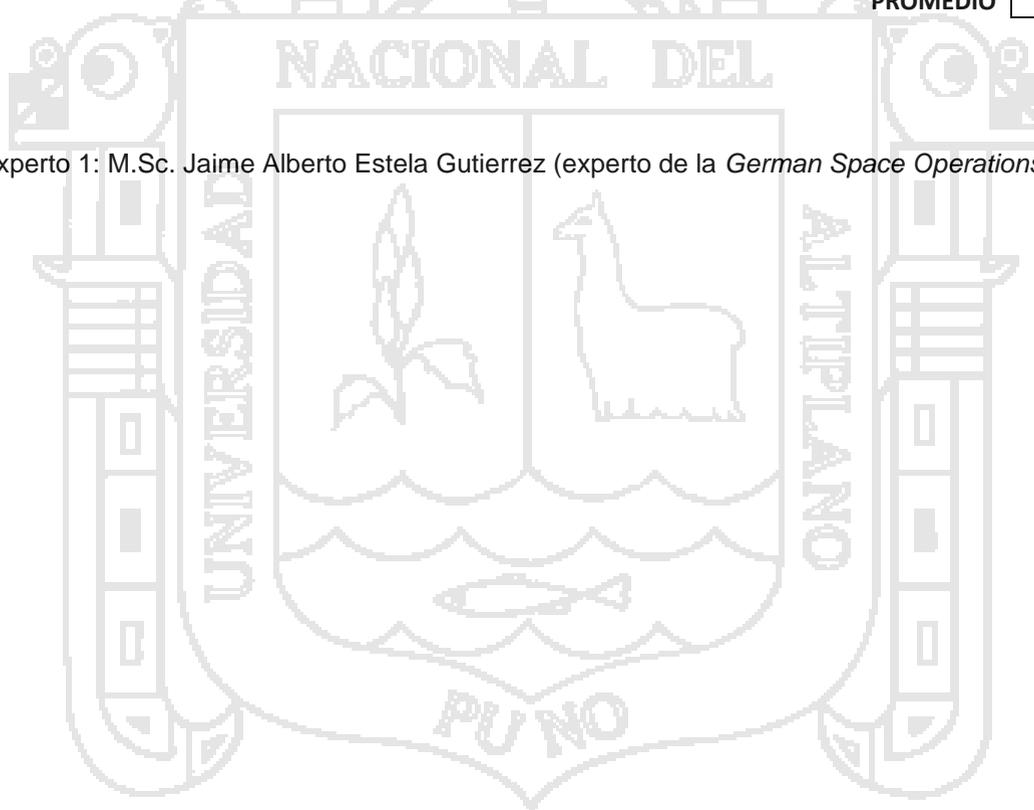
Ciudad, 27 de agosto de 2014

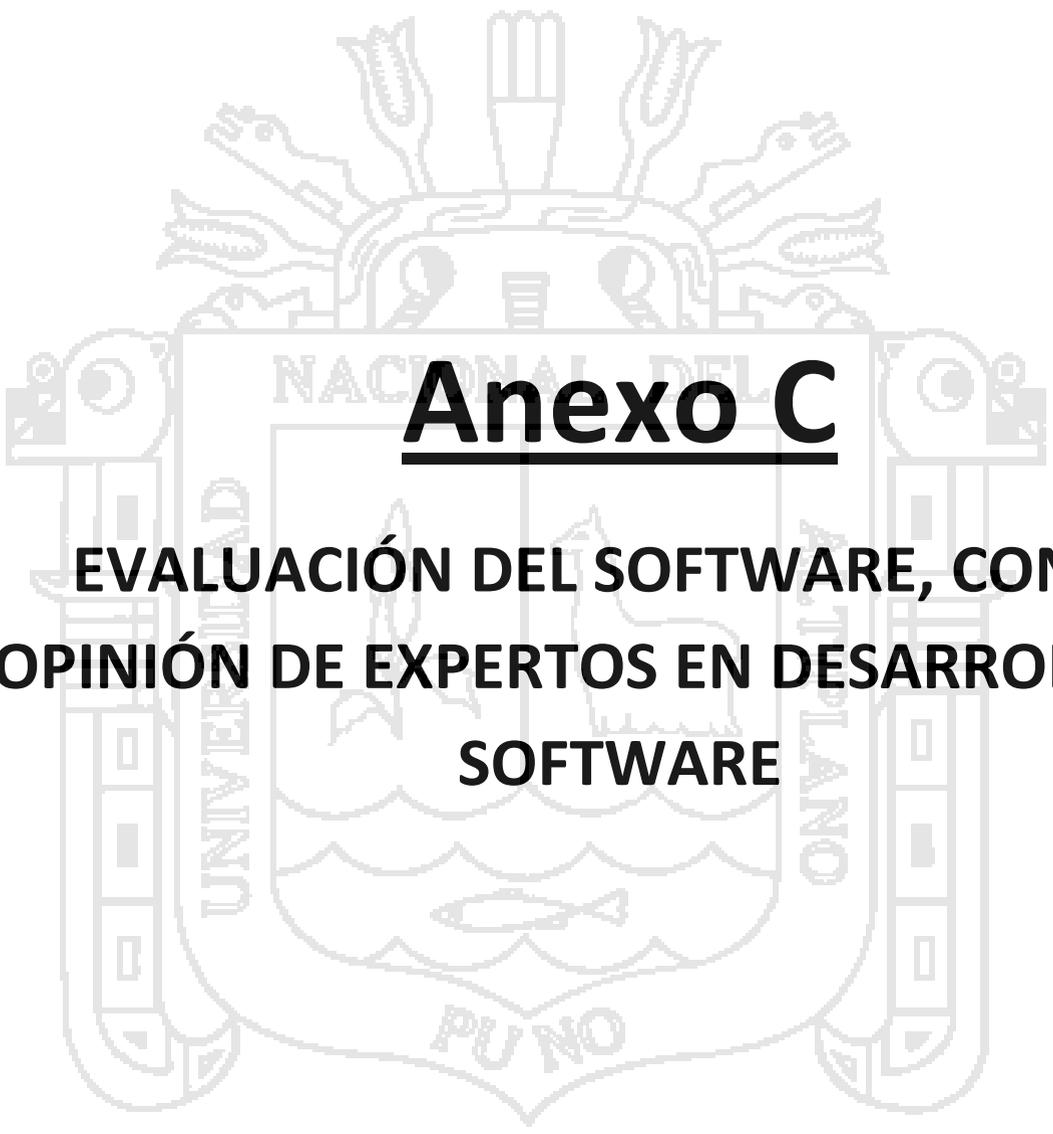
.....
Firma del Experto Informante

TABLA DE EVALUACIÓN DE INSTRUMENTOS POR EXPERTOS

N°	Indicadores	Criterios	Expertos
			1
1	Claridad	Esta formulado con lenguaje apropiado	75
2	Objetividad	Esta expresado con conductas observables	75
3	Actualidad	Adecuado al avance de la ciencia	80
4	Organización	Existe una organización lógica	75
5	Suficiencia	Comprende los aspectos en cantidad y calidad	75
6	Intencionalidad	Adecuado para valorar la gestión tecnológica	85
7	Consistencia	Basado en aspectos teóricos científicos	85
8	Coherencia	Entre los índices e indicadores	75
9	Metodología	La estrategia responde al propósito del diagnóstico.	80
10	Pertinencia	Es útil y adecuado para la investigación.	90
PROMEDIO			79.5

Experto 1: M.Sc. Jaime Alberto Estela Gutierrez (experto de la *German Space Operations Center*).





Anexo C

EVALUACIÓN DEL SOFTWARE, CON LA OPINIÓN DE EXPERTOS EN DESARROLLO DE SOFTWARE



INFORME DE EVALUACIÓN DE EXPERTOS EN SOFTWARE

INFORME DE EVALUACIÓN DE EXPERTOS DE SOFTWARE

I. DATOS GENERALES:

Apellidos y Nombres del Informant Ing. José Maquera
Institución donde labora: Trans Solutions Systems
Nombre del Software: Software de Telecontrol y Telemetría.

II. INSTRUCCIONES.

A continuación se presenta un conjunto de proposiciones que miden las características para evaluar la calidad de un software según la norma ISO/IEC 9126.

Para cada proposición marque con una "X" el casillero correspondiente según el grado de apreciación merecida de acuerdo a la siguiente escala, donde:

1-Deficiente 2-Insuficiente 3-Aceptable 4-Sobresaliente 5-Excelente

De lo cual, de:

0 a 30 puntos, equivale al cumplimiento de características: Nula.

31 a 50 puntos, equivale al cumplimiento de características: Poco.

51 a 70 puntos, equivale al cumplimiento de características: Regular.

71 a 90 puntos, equivale al cumplimiento de características: Bastante.

91 a 100 puntos, equivale al cumplimiento de características: Completamente.

III. ASPECTOS DE VALIDACIÓN:

INDICADORES	CRITERIO	PLANTEAMIENTO	APRECIACIÓN					
			1	2	3	4	5	
Funcionalidad	Adecuación	1. Tiene funciones apropiadas para las tareas especificadas.						X
	Exactitud	2. Hace lo que fue acordado en forma esperada y correcta.						X
	Interoperabilidad	3. Interactúa con otros sistemas especificados.	X					
	Seguridad	4. Es capaz de proteger la información y datos.	X					
Fiabilidad	Madurez	5. Se presenta fallas por defectos y errores.						X
	Recuperabilidad	6. Es capaz de recuperar su funcionamiento adecuado y datos afectados en caso de fallas.					X	
	Tolerancia a Fallos	7. Mantiene el nivel de funcionamiento en caso de fallos.						X
Usabilidad	Aprendizaje	8. Es fácil de aprender a usar.						X
	Comprensión	9. Es fácil de entender y reconocer la estructura, la lógica y su aplicabilidad.						X
	Operatividad	10. Es fácil de operar y utilizar.						X
	Atractividad	11. Es atractivo el diseño.			X			
Eficiencia	Comportamiento en el tiempo	12. El tiempo de respuesta y procesamiento es suficientemente rápido.						X
	Comportamiento de Recursos.	13. Es capaz de utilizar cantidades y tipos adecuados de recursos.					X	



	14. Hay riesgo o efectos inesperado mientras esta en uso.						x
	15. Es fácil identificar una falla.						x
	16. Es fácil de modificar y adaptar.	x					
	17. Es fácil y sencilla la instalación en el contexto esperado.						x
	18. Es fácil de reemplazarlo por otra aplicación similar.						x
	19. Es fácil de adaptarse en otras plataformas (S.O.)						x
	20. Comparte sin problema recursos con otros aplicativos o dispositivos.						x
84							

IV. OBSERVACIONES AL SOFTWARE:

V. SUGERENCIAS:

VI. CALIFICACIÓN

Cumple con las características: *BASTANTE*

Puno, 04 de Agosto de 2014.

.....
Firma del Experto Informante