



FACULTAD DE CIENCIAS E INGENIERÍA

INGENIERÍA DE SISTEMAS E INFORMÁTICA

TEMA:

**“Evaluar el aseguramiento de la calidad de software
para el Proyecto Sistema Colegio.”**

CURSO DE:

Calidad de Software

Mg. Guevara Jiménez, Jorge Alfredo

ESTUDIANTES:

Huaraz Morales, Shalom Adonai

Cod 12110040

Paucar Carrasco, Max Iván

Cod 11101133

Lima – Perú

2021- I

DEDICATORIA

A nuestros padres por la gran formación que nos dieron, por hacernos las personas de bien que somos en la actualidad; muchos de nuestros logros se los debemos a ellos. Gracias por motivarnos constantemente para alcanzar nuestros anhelos.

AGRADECIMIENTO

A nuestras familias por hacer posible el estudio de nuestra carrera, para nuestro grupo de estudio, por fortalecer nuestras dificultades en algunas materias, a la dirección y comité del centro educativo, así mismo al Mg. Guevara Jiménez, Jorge Alfredo por su asesoramiento haciendo posible la elaboración del presente proyecto de software Sistema Colegio – sistema de escritorio y a nuestros compañeros del IX ciclo de ingeniería de sistemas por su sugerencia y aprendizaje mutuo.

CONTENIDO

RESUMEN EJECUTIVO	12
CAPÍTULO I. INTRODUCCIÓN	14
1.1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL TEMA	14
1.2. SITUACIÓN ACTUAL.....	14
1.3. PROBLEMA.....	15
1.4. JUSTIFICACIÓN	15
1.5. OBJETIVOS DE LA INVESTIGACIÓN	15
1.5.1. <i>Objetivo general</i>	15
1.5.2. <i>Objetivo específico</i>	15
1.6. METODOLOGÍA DE LA INVESTIGACIÓN.....	16
1.7. LIMITACIONES	16
1.8. ALCANCES DE LA INVESTIGACIÓN	17
2.1. CICLO DE VIDA DEL PROYECTO - PMBOK®.....	18
2.2. TEMÁTICAS DE LAS ÁREAS DE CONOCIMIENTO DE LA DIRECCIÓN DE PROYECTOS	19
2.3. MARCO DE TRABAJO	20
2.4. INGENIERÍA HACIA EL PRODUCTO - TÉCNICAS Y HERRAMIENTAS.....	20
2.4.1. <i>Enfoque de cascada</i>	21
2.4.2. <i>Enfoque incremento</i>	22
2.5. CALIDAD DE SOFTWARE.....	22
2.5.1. <i>Calidad de los procesos del software</i>	23
2.5.2. <i>Calidad del producto de software</i>	23
2.5.3. <i>Modelo de Calidad Interna y Externa</i>	25
2.5.4. <i>Modelo para la calidad en uso para el producto software</i>	29
2.5.5. <i>Prueba de caja negra</i>	30
2.5.6. <i>Prueba de caja blanca</i>	31
2.5.7. <i>Cobertura de código</i>	32
CAPÍTULO III. INGENIERÍA DEL PRODUCTO O DE DESARROLLO DE LA INVESTIGACIÓN	33
3.1. ETAPA DE INICIO	33
3.1.1. <i>Descripción de la empresa</i>	33
3.1.2. <i>Descripción global del software</i>	34
3.2. ETAPA DE PLANIFICACIÓN	39
3.2.1. <i>Desarrollo de principios de calidad de software</i>	39
3.2.2. <i>Desarrollo de modelos de calidad del proceso</i>	43
3.2.3. Desarrollo de modelos de calidad del producto.	89
a) <i>Hoja de comprobación de modelos de calidad del producto.</i>	89
3.3. ETAPA DE EJECUCIÓN.....	92
3.3.1. Evaluación de calidad del proceso.	92
3.3.2. <i>Evaluación de calidad del producto</i>	108
3.3.3. <i>Desarrollo de pruebas de software.</i>	120
3.4. PRUEBAS DE CAJA NEGRA Y CAJA BLANCA	120
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES	121
4.1. CONCLUSIONES.....	121
4.2. RECOMENDACIONES	122
REFERENCIAS BIBLIOGRÁFICAS	123
ÍNDICE DE ABREVIATURAS Y SIGLAS.....	124
GLOSARIO DE TÉRMINOS	125
APÉNDICES.....	129

ÍNDICE DE TABLAS

Tabla 1. Áreas de conocimiento	19
Tabla 2. Requisitos del Sistema.	34
Tabla 3. Casos de uso.	35
Tabla 4. Modelos de calidad del producto.	89
Tabla 5. Evaluación de calidad del proceso.	92
Tabla 6. Nivel de adherencia en la evaluación de calidad del proceso.	94
Tabla 7. Calidad de uso del producto.	108
Tabla 8. Nivel de adherencia en la calidad en uso.	109

ÍNDICE DE FIGURA

Figura 1. Etapas de un proyecto (Guía PMBOK)	18
Figura 2. Enfoque de cascada.....	21
Figura 3. Enfoque incremento.	22
Figura 4. Ciclo de vida de la calidad del producto software. (ISO/IEC 25000)	24
Figura 5. Modelo de calidad interna y externa. (ISO/IEC 25010)	25
Figura 6. Modelo para la calidad de uso para producto software. (ISO/IEC 25010).....	29
Figura 7. Organigrama del Colegio Matemático Honores de Zapallal	34
Figura 8. Diagrama de uso.	36
Figura 9. Programación por Capas.	37
Figura 10. Arquitectura de Hardware	38
Figura 11. Principios de Ingeniería de Software.....	39
Figura 12. Principios de Gestión de Proyectos.....	40
Figura 13. Principios de Código Fuente	41
Figura 14. Modelo GQM.....	42
Figura 15. Calidad del Proceso de Ingeniería de Software	43
Figura 16. Caso de uso	44
Figura 17. Caso de uso de negocio.....	44
Figura 18. Diagrama de casos de uso.....	45
Figura 19. Actores del sistema.	45
Figura 20. Casos de uso del sistema.	46
Figura 21. Diagramas de caso de uso de sistema.	47
Figura 22. Requerimientos funcionales del sistema.	47
Figura 23. Requerimientos no funcionales del sistema.	48
Figura 24. Requerimientos para la implementación del sistema.	48
Figura 25. Diagrama de actividades.....	48
Figura 26. Diseño de módulos del sistema.....	49
Figura 27. Marcos de trabajo necesarios para la implementación del sistema.....	49
Figura 28. Diseño de la interfaz.....	50
Figura 29. Diagrama de colaboración.	50
Figura 30. Diagrama de secuencias.....	51
Figura 31. Modelo de la base de datos.	51
Figura 32. Programación de interfaces	52
Figura 33. Modelo de despliegue del sistema.	53
Figura 34. Proceso de desarrollo scrum.....	53
Figura 35. Arquitectura del sistema.	53
Figura 36. Pantalla principal del sistema.....	54
Figura 37. Panel de control.	54
Figura 38. Módulo de alumnos.	55
Figura 39. Módulo de apoderado.	55
Figura 40. Modulo aulas.....	55
Figura 41. Modulo notas.....	56
Figura 42. Modulo matriculas.	56
Figura 43. Modulo docente.....	57
Figura 44. Modulo año escolar.....	57

Figura 45. Calidad de proceso de gestión.	61
Figura 46. Acta de constitución del proyecto PSC.	61
Figura 47. Evidencia documentación del proyecto.	62
Figura 48. Diagrama de caso de uso.	62
Figura 49. Requerimientos iniciales del sistema.	63
Figura 50. Alcance del proyecto.	63
Figura 51. EDT.	64
Figura 52. Cronograma del PSC.	64
Figura 53. Actividades del negocio.	65
Figura 54. Diagrama de secuencias actualizar matricula.	65
Figura 55. Duración de las actividades.	66
Figura 56. Cronograma del proyecto.	66
Figura 57. Plan de gestión de comunicaciones PSC.	67
Figura 58. Plan de Comunicaciones.	67
Figura 59. Gestión de la calidad.	68
Figura 60. Plan de respuesta de control de riesgos.	69
Figura 61. Plan de gestión de adquisiciones.	70
Figura 62. Plan de Gestión de los interesados.	70
Figura 63. Plan de acción de control de riesgos.	71
Figura 64. Definición de roles del proyecto.	72
Figura 65. Identificación de riesgos.	73
Figura 66. Análisis cualitativo de riesgos.	73
Figura 67. Análisis cuantitativo de riesgos.	75
Figura 68. Realización del control de riesgos.	76
Figura 69. Documentación del proyecto en GitHub.	77
Figura 70. Hoja de comprobación de calidad de código fuente.	78
Figura 71. Líneas de comentarios de código.	79
Figura 72. Comentario de código por módulo.	79
Figura 73. Comentario de código por módulo.	79
Figura 74. Inicialización de variables.	80
Figura 75. Cota superior de arreglos.	80
Figura 76. Uso de delimitador de cadenas.	81
Figura 77. Posibilidad de desbordamiento de buffer.	81
Figura 78. Uso condicional.	81
Figura 79. Certeza de terminación de un ciclo.	82
Figura 80. Enunciados correctamente colocados en paréntesis.	82
Figura 81. Enunciados justificados.	83
Figura 82. Uso de paréntesis en enunciados.	83
Figura 83. Uso de variables de entrada.	83
Figura 84. Asignación de valores a las variables.	84
Figura 85. Entradas inesperadas.	84
Figura 86. Uso de variables de entrada.	84
Figura 87. Uso de valor a las variables de salida.	85
Figura 88. Entradas inesperadas causan corrupción.	85
Figura 89. Llamada de función, métodos y parámetros	86
Figura 90. Tipos de parámetros formales y reales coinciden.	86
Figura 91. Uso de parámetros en el orden correcto.	86
Figura 92. Modelo de estructura de memoria compartida.	87

Figura 93. Modelo de estructura de memoria compartida.	87
Figura 94. Posibles condiciones de error a tomar en cuenta.....	88
Figura 95. Script de la creación de la base de datos.....	88
Figura 96. Cumplimiento de requerimientos funcionales.....	110
Figura 97. Cumplimiento de objetivos.	110
Figura 98. Cumplimiento de tareas terminadas.....	111
Figura 99. Frecuencia de errores.	111
Figura 100. Tiempo para completar una tarea.	112
Figura 101. Eficiencia de los usuarios.	112
Figura 102. Eficiencia del usuario.	112
Figura 103. Acciones productivas del usuario.	113
Figura 104. Productividad del usuario	113
Figura 105. Cumplimientos de tareas.....	113
Figura 106. Satisfacción del usuario.	114
Figura 107. Satisfacción del usuario.	114
Figura 108. Funcionalidad intuitiva.....	115
Figura 109. Eficiencia del software.....	115
Figura 110. Software operativo en producción.	116
Figura 111. Grado de incidencia de corrupción de software.	116
Figura 112. El software es amigable con el usuario.	116
Figura 113. Software de fácil uso.	117
Figura 114. Riesgos por uso de software.....	117
Figura 115. Automatización de registros.	117
Figura 116. Integridad del software.	118
Figura 117. Software de diseño modular.....	118
Figura 118. Flexibilidad para la implementación.	119

CERTIFICADO DE REVISIÓN DE REDACCIÓN Y GRAMÁTICA

Documento: G.P-001

Nosotros, Shalom Adonai Huaraz Morales y Max Paucar Carrasco certificamos: que revisamos la redacción y ortografía del contenido del proyecto de investigación: “Proyecto Sistema Colegio – Sistema Escritorio.”

Para el efecto he procedido a leer y a analizar de manera profunda el estilo y la forma del contenido del texto:

- Se denota la claridad en la escritura en todas sus partes.
- La acentuación es toda precisa.
- Se utilizan los signos de puntuación de manera acertada.
- En todos los ejes temáticos se evitan los vicios de dicción.
- Hay concreción y exactitud en las ideas.
- No incurrir en errores en la utilización de las letras.
- La aplicación de la sinonimia es correcta.
- Se maneja con conocimiento y precisión la morfosintaxis.
- El lenguaje es pedagógico, académico, sencillo y directo, por lo tanto, de fácil comprensión.

Por lo expuesto, y en uso de nuestros derechos como estudiantes, recomendamos la VALIDEZ ORTOGRÁFICA del proyecto previo a la presentación y evaluación del profesor.

Atentamente,



Shalom Adonai Huaraz Morales



Max Iván Paucar Carrasco

DECLARACIÓN DE AUTENTICIDAD Y DE NO PLAGIO

Nosotros, Huaraz Morales, Shalom Adonai identificado con D.N.I. 48123639, Paucar Carrasco, Max Iván identificado con D.N.I. 01343134 estudiantes de la FCI-UCH, autor(a/es) del proyecto de investigación: “Proyecto Sistema Colegio – Sistema Escritorio.”

DECLARAMOS QUE:

1. El presente trabajo de investigación, tema presentado para la aprobación del curso es original, siendo resultado de nuestro trabajo personal, el cual no hemos copiado de otro trabajo de investigación, ni utilizado ideas, fórmulas, ni citas completas “stricto sensu”; así como ilustraciones diversas, sacadas de cualquier tesis, obra, artículo, memoria, etc., (en versión digital o impresa). Caso contrario, mencionó de forma clara y exacta su origen o autor, tanto en el cuerpo del texto, figuras, cuadros, tablas u otros que tengan derechos de autor.
2. Declaramos que el trabajo de investigación que ponemos en consideración para evaluación no ha sido presentado anteriormente para obtener algún grado académico o título, ni ha sido publicado en sitio alguno. Somos conscientes de que el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales, por lo que asumimos cualquier responsabilidad que pudiera derivarse de irregularidades en la tesis, así como de los derechos sobre la obra presentada. Asimismo, nos hacemos responsable ante la universidad o terceros, de cualquier irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado. De identificarse falsificación, plagio, fraude, o que el trabajo de investigación haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándose por todas las cargas pecuniarias o legales que se deriven de ello sometiéndose a la normas establecidas y vigentes de la UCH.

Los Olivos, 20 de abril del 2021



Shalom Adonai Huaraz Morales



Max Iván Paucar Carrasco

RESUMEN EJECUTIVO

El presente proyecto es un trabajo de investigación aplicada sobre la evaluación de la calidad de un producto de software. Los sistemas de información son herramientas de mayor utilidad para la optimización de procesos en las organizaciones de todo nivel y tamaño, con el propósito de eficiencia y satisfacción de necesidades, el software debe contar con criterios que garanticen su calidad. El objetivo es analizar los estándares relacionados a la calidad del producto, calidad del proceso y calidad de gestión en el Proyecto Sistema Colegio. Durante la realización de la investigación se tenía conocimiento sobre la documentación de los procesos del sistema que forman parte del proceso de desarrollo. El proceso de aseguramiento de la calidad de software es parte fundamental de los procesos de desarrollo modernos, donde hay la necesidad de cuidar la calidad de los productos terminados que se desarrollan, de ahí la importancia de conformando los procesos de aseguramiento de la calidad de software, donde se involucran tres aspectos fundamentales, personas, procesos y herramientas, mismas que se unen para permiten gestionar la calidad durante el desarrollo de un producto de software, importante tomar en cuenta durante la definición de una área de aseguramiento de la calidad es lograr la integración de los tres aspectos mencionados para con ello facilitar la conformación y buen desarrollo del área. Importante es la formación académica para conocer las técnicas de aseguramiento de la calidad de software para que las organizaciones y con ello ampliar la oportunidad de adoptar nuevas tecnologías que ayuden a innovar y fortalecer sus áreas de desarrollo de software.

Palabras clave: Calidad del software; evaluación; mantenibilidad; aseguramiento de calidad; proceso.

ABSTRACT

This project is an applied research work on the evaluation of the quality of a software product. Information systems are the most useful tools for the optimization of processes in organizations of all levels and sizes, with the purpose of efficiency and satisfaction of needs, the software must have criteria that guarantee its quality. The objective is to analyze the standards related to product quality, process quality and management quality in the Sistema Colegio Project. During the research, the documentation of the system processes that are part of the development process was known. The software quality assurance process is a fundamental part of modern development processes, where there is the need to take care of the quality of the finished products that are developed, hence the importance of conforming the software quality assurance processes, where three fundamental aspects are involved, people, processes and tools, which come together to manage the quality during the development of a software product, important to take into account during the definition of an area of quality assurance is to achieve the integration of the three aspects mentioned to thereby facilitate the conformation and good development of the area. It is important to have the academic training to know the techniques of software quality assurance for organizations and thus expand the opportunity to adopt new technologies that help to innovate and strengthen their software development areas.

Keywords: *Software quality; testing; maintainability; quality assurance; process.*

CAPÍTULO I. INTRODUCCIÓN

1.1. Planteamiento y justificación del tema

En la actualidad en la industria del software, el aseguramiento de calidad del producto es un proceso crítico en el desarrollo de software. Por ello hace uso de técnicas que permitan gestionar el proceso de aseguramiento de calidad en el desarrollo de software. El presente proyecto será un estudio de la calidad del Proyecto Sistema Colegio, implementado en el colegio Honores del Zapallal, como objetivo general del proyecto es desarrollar una propuesta integral que permita el aseguramiento de calidad en el proceso del desarrollo de software y mejorar su gestión, haciendo uso de los estándares preestablecidos, técnicas y métodos para el desarrollo de software de la ISO/IEC 25000, donde menciona el establecimiento de un marco de trabajo común para evaluar la calidad de un producto de software, para establecer métricas de calidad e informar los resultados del proceso mediante indicadores. Los procesos de matrículas y control de notas que se llevan a cabo actualmente se realizan de forma manual, de tal manera que el proceso de matrícula se demora en la gestión debido a la cantidad de datos que hay que ingresar por alumno implicando menos avance en todo el proceso y un mayor tiempo a lo planificado en la institución; haciendo uso de respectivos recursos como folders, carpetas y hojas de registro las cuales por la seguridad de la institución no se consideran muy seguras porque con el tiempo de los años estas se deterioran además son muy frágiles ante cualquier eventualidad natural, todo esto implica un elevado número de inconvenientes como control adecuado en el cobro de las pensiones por parte de la administración provocando una pérdida económica para la institución educativa.

1.2. Situación actual

El Colegio Matemático Honores de Zapallal debido al crecimiento institucional, se encuentra en un proceso de sistematizar la información para ser procesada, almacenada con el objetivo de agilizar el control de notas y el proceso de matrícula de los alumnos. El desarrollo del sistema escritorio ayudó a solucionar las necesidades del departamento de secretaria y una mayor relación entre directivos, profesores y estudiantes. Proporcionando una mejor efectividad en el manejo del flujo de los datos y documentos de los

estudiantes, y al mismo tiempo facilitando el manejo de información que representa un recurso viable para que la información sea ágil, eficiente y eficaz cuando sea requerida.

1.3. Problema

En la institución educativa existe la necesidad de hacer un estudio de aseguramiento de calidad en el desarrollo del sistema informático para el proceso de matrícula y control de notas del colegio.

1.4. Justificación

La principal finalidad es evaluar las especificaciones y el cumplimiento de las características del producto con una propuesta integral de aseguramiento de la calidad de software con la norma ISO 25000 en el Sistema de Matrícula y Control de Notas para el Colegio Matemático Honores de Zapallal, identificando errores y evitando que estos aparezcan una vez entregado al usuario final, el software beneficiará de manera directa a la Secretaría Académica, los profesores y estudiantes. Este estudio está orientado a verificar el cumplimiento de las características que permitan alcanzar el nivel de calidad no solo en el proceso de desarrollo del software, también en la gestión del producto y la documentación para alcanzar la satisfacción del cliente.

1.5. Objetivos de la investigación

1.5.1. Objetivo general

Evaluar un estudio integral que permita un aseguramiento de calidad de software para el sistema escritorio del Colegio Matemático Honores del Zapallal.

1.5.2. Objetivo específico

- Hacer una revisión en la gestión de calidad de proceso.
- Verificar las actividades de aseguramiento de la calidad del producto de acuerdo a lo establecido.
- Realizar las pruebas de calidad externa y calidad en uso.
- Medir e informar los resultados de las revisiones de aseguramiento de control de la calidad.
- Identificar y proponer mejoras a partir de las revisiones realizadas.

1.6. Metodología de la investigación

En la presente investigación se aplicará la Metodología de Desarrollo RUP (*The Rational Unified Process*). Esta metodología es un marco de trabajo con enfoque a la disciplina que asigna tareas y responsabilidades en el proceso, su objetivo es garantizar la calidad del producto satisfaciendo las necesidades del cliente, realizando una correcta estimación de los tiempos, en llevar una documentación exhaustiva de todo el proyecto, la planificación y control del mismo, en especificaciones precisas de requisitos, modelado y en cumplir con un plan de trabajo, definido todo esto, en la fase inicial del desarrollo del proyecto. RUP es un conjunto de metodologías adaptables al contexto y necesidades de cada organización. Es el resultado de varios años de desarrollo de técnicas de desarrollo, a través del UML.

Características Principales de RUP

- Usado para desarrollar grandes y complejos proyectos.
- Orientado a Objetos.
- Utiliza UML

Características del ciclo de vida de RUP

- Dirigido por casos de uso para modelar el negocio a través de los requerimientos.
- El modelo de arquitectura se representa a través de vistas de diagramas de UML.
- Las iteraciones son pasos en los flujos de trabajo, y los incrementos en el crecimiento del producto.

Diferencias de RUP con las demás metodologías

Los casos de uso especifican los requisitos del sistema, el diseño, implementación y prueba. Los cambios posteriores durante la construcción y el mantenimiento. (Maida & Pacienza, 2015)

1.7. Limitaciones

El estudio se limita al proceso de matrículas y consulta de notas bajo la norma ISO 25000.

La evaluación se realizará en un ambiente controlado de prueba.

1.8. Alcances de la investigación

Se realizará un control de la calidad de software al sistema de matrícula y notas del Colegio Matemático Honores del Zapallal. Las Métricas utilizadas para medir los resultados obtenidos, sino también para poder mejorar los procesos internos. El Método aplicado en el caso de estudio, para la evaluación del proceso integral en el desarrollo de una aplicación escritorio, ayudando a controlar de mejor manera los cambios y futuros riesgos que podrían ocurrir en la implementación y puesta en operaciones del sistema informático.

CAPÍTULO II. MARCO TEÓRICO Y CONCEPTUAL

2.1. Ciclo de vida del proyecto - PMBOK®

El Project Management Institute (PMI) con la guía de referencia para todo Project Manager El PMBOK® Guide define los estándares para los proyectos puedan adaptarse a cada caso y contexto particular como “un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único”. Por tanto, el ciclo de vida del proyecto son las distintas fases por las que atraviesa el mismo, desde su inicio hasta su conclusión. Estos ciclos proporcionan el marco de referencia para dirigir el proyecto.

Al comienzo de un proyecto, la cantidad de planificación y trabajo requeridos es abrumadora. Por eso los gerentes experimentados dividen un proyecto en fases del proyecto son divisiones dentro del mismo proyecto, donde es necesario ejercer un control adicional para gestionar eficazmente la conclusión de un entregable mayor. Dividir en fases ayuda a estructurarlo y simplificarlo en una serie de pasos lógicos y manejables. Las fases del proyecto suelen completarse de manera secuencial, pero en determinadas situaciones de un proyecto pueden superponerse. (Barato, 2015. p. 17)

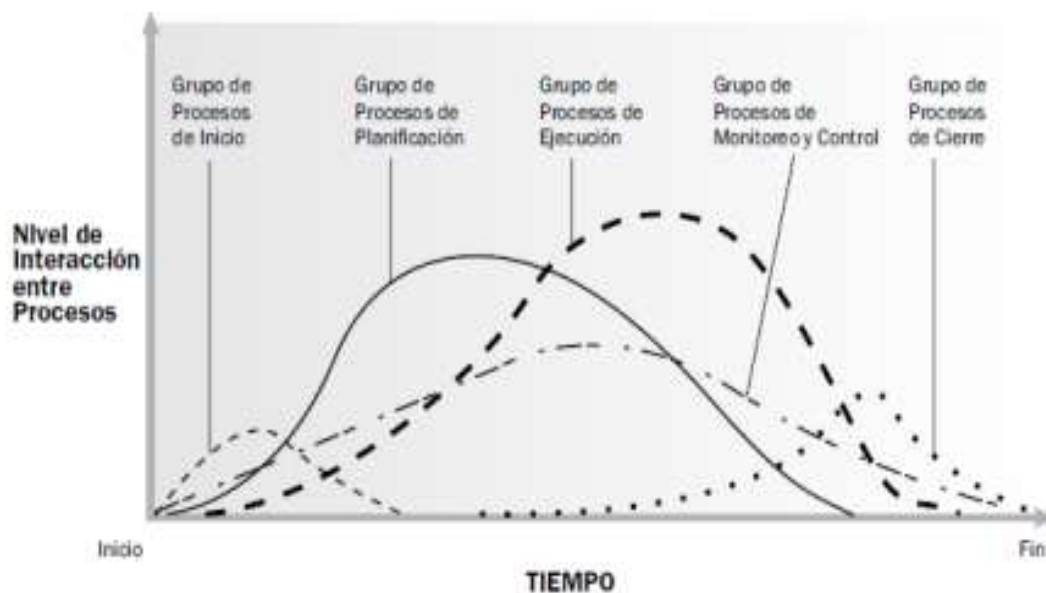


Figura 1. Etapas de un proyecto (Guía PMBOK)

Según el PMBOK, un proyecto es el trabajo que se realiza para crear un producto, y es temporal teniendo un inicio y un final establecidos, y que finaliza en cuanto se logran los objetivos del proyecto. Será mediante la buena gestión y dirección de proyectos, con la ayuda de técnicas, herramientas y

especificaciones del producto se lograrán finalizar el proyecto. Los puntos principales a considerar son:

- Resolver todas las inquietudes, necesidades y expectativas de los involucrados según la planificación y ejecución del proyecto.
- Realizar un organigrama de la empresa, plantear objetivos y políticas con objetivo de tener amplia visión de la organización.
- Plantear y relacionar las limitaciones y riesgos del proyecto con el alcance, calidad, cronograma, presupuesto, recursos y riesgo en base a los procesos de la organización.

2.2. Temáticas de las áreas de conocimiento de la dirección de proyectos

La Guía del PMBOK por medio de diferentes gestiones (integración, alcance, cronograma, costos, calidad, recursos, comunicaciones y riesgos) nos indica cómo gestionar un proyecto separando jerárquicamente los trabajos a ejecutar por parte de los responsables y se orienta a los entregables requeridos para lograr los objetivos del proyecto.

La división del trabajo se realiza en una Estructura Desglosable de Trabajo (EDT), donde se organiza y precisa el alcance total de todos los documentos entregables exponiendo la definición detallada de todas las entradas de información, los procesos y salidas de documentación e información que demanda.

Tabla 1. Áreas de conocimiento

Gestión de la integración	Implica la toma de decisiones referidas a la asignación de los recursos, balancear los objetivos y entre las áreas de conocimiento manejar sus interdependencias.
Gestión del alcance	Garantiza que el proyecto cuente con todo lo necesario para completarlo, incluyendo los procesos requeridos en el proyecto. Su principal objetivo es definir y controlar qué se incluye y no se incluye en el proyecto
Gestión del tiempo	Administra los procesos necesarios para la finalización del proyecto a tiempo. Los procesos que incorpora son: Definición de las actividades, establecer las secuencias de las actividades, estimar los recursos de las actividades, programar la duración de las actividades, y desarrollar y controlar el cronograma.
Gestión de los costos	Contiene los procesos de estimar, presupuestar y controlar los costos, con la finalidad de que el proyecto se ejecute con el presupuesto aprobado.
Gestión de la calidad	Aquí se determinan responsabilidades, objetivos y políticas de calidad, para que el proyecto se ejecute satisfactoriamente.

Gestión de los recursos humanos	Aquí está la organización, gestión y conducción del equipo del proyecto. Este equipo está conformado por personas a quienes se les asigna sus roles y responsabilidades para completar el proyecto.
Gestión de las comunicaciones	Aquí se busca que la generación, recopilación, distribución, almacenamiento, recuperación y disposición final de la información del proyecto sean oportunos y adecuados.
Gestión de los riesgos	Se desarrolla la planificación de la gestión, la identificación, el análisis, la planificación de respuestas a los riesgos, así como su monitoreo, control y minimización en un proyecto.
Gestión de las adquisiciones	Se abarca los procesos de compra o adquisición de los insumos, bienes y servicios que se requiere para hacer realidad el proyecto.
Gestión de los interesados	Se desarrollan los procesos que hacen posible la identificación de las personas, grupos u organizaciones que puedan ser afectados o no en el proyecto. Se busca conocer y evaluar las expectativas de los interesados y su impacto en el proyecto.

2.3. Marco de trabajo

Las metodologías son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, con una flexibilidad para moldear el proyecto y su desarrollo a las circunstancias específicas del entorno. Las habilidades dentro del trabajo que van teniendo las personas están en constante progreso y retroalimentación de las lecciones aprendidas en las diferentes actividades de trabajo reduciendo los costes e incrementando la productividad. (Villán, Vanessa 2019).

A continuación, enumeramos algunas de las ventajas que nos brinda la gestión de proyectos:

- Mejora de la calidad del producto.
- Mayor satisfacción del cliente.
- Mayor motivación de los trabajadores.
- Trabajo colaborativo.
- Mayor control y capacidad de predicción.
- Reducción de costes.

2.4. Ingeniería hacia el producto - técnicas y herramientas

Para el proceso del desarrollo en la ingeniería del producto tendremos 2 enfoques tomados como técnicas y herramientas, para el desarrollo del producto del software, que lo detallaremos a continuación:

2.4.1. Enfoque de cascada.

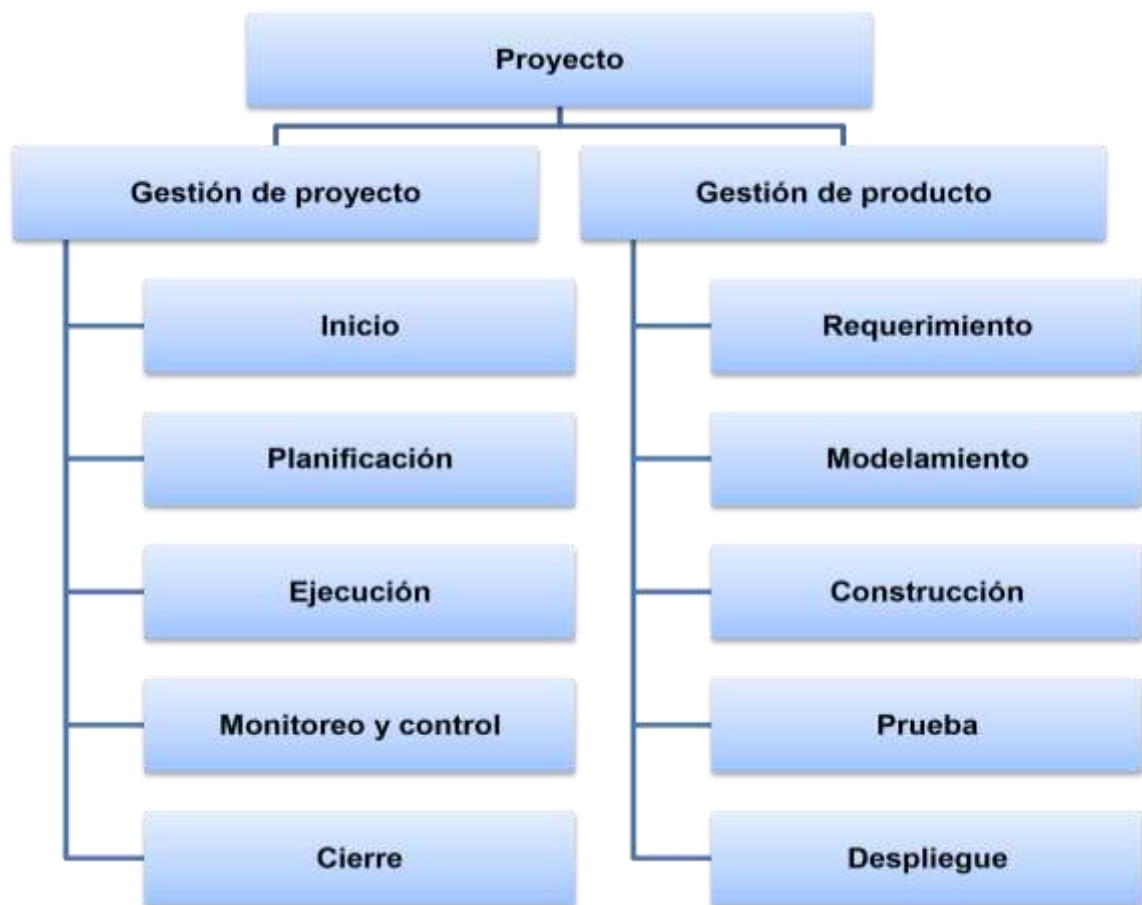


Figura 2. Enfoque de cascada.

Este enfoque metodológico de procedimiento lineal que se desarrolla en forma secuencial, en el que el desarrollo de software se concibe como un conjunto de etapas que se ejecutan una tras otra. Cada fase se ejecuta una sola vez. Se le denomina así por las posiciones que ocupan las diferentes fases que componen el proyecto, colocadas una encima de otra, y siguiendo un flujo de ejecución de arriba hacia abajo, como una cascada. (Almeida, 2015)

2.4.2. Enfoque incremento.

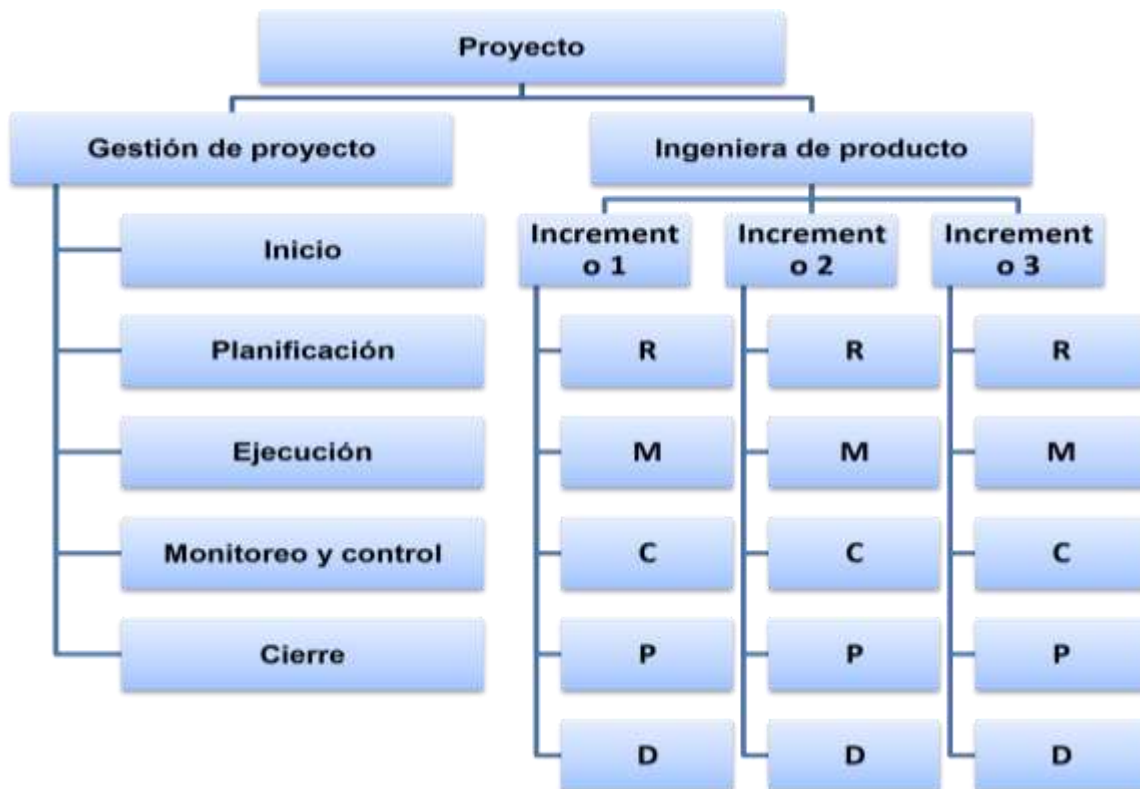


Figura 3. Enfoque incremento.

El modelo incremental de gestión de proyectos tiene como objetivo un crecimiento progresivo de la funcionalidad. Es decir, el producto va evolucionando con cada una de los entregables previstos hasta completar el total del producto que se amolda a lo requerido por el cliente. Este enfoque se usó inicialmente para proyectos de software, para también aplicarse a otros sectores. Los responsables del proyecto deben analizar si los resultados parciales son los esperados. (Almeida, 2015)

2.5. Calidad de software.

Es el grado en que el producto software incorpora un conjunto de características, definidas por la industria, de tal manera que se garantiza su eficiencia de uso, respecto a los requerimientos de los clientes. Las implicaciones de la calidad del software son:

- Métricas
- Inspecciones
- Pruebas
- Procesos.

Estos son aplicados en el ciclo de desarrollo de un proyecto. Es decir, calidad de software, implica evaluar dos ámbitos: el producto final y los procesos.

2.5.1. Calidad de los procesos del software.

Es el conjunto estructurado de actividades requeridas para desarrollar un sistema de software, los cuales son: especificaciones, diseño, validación, evolución, desarrollo y mantenimiento.

Los procesos que se desarrollan son:

- a) Proceso de implementación y cambios.
 - Infraestructura de procesos
 - Ciclo de gestión de los procesos de software
 - Modelos para el proceso de implementación y cambio
 - Consideraciones prácticas
- b) Definición de procesos
 - Modelos de ciclo de vida del software
 - Procesos de ciclo de vida del Software
 - Modelos para el proceso de implementación y cambio
 - Adaptaciones y automatización
- c) Evaluación de procesos
 - Modelos de evaluación del proceso
 - Métodos de evaluación del proceso
- d) Medidas de productos y procesos
 - Medición del proceso
 - Medición de productos de software
 - Calidad de los resultados de la medición
 - Modelos de información de software

2.5.2. Calidad del producto de software

El modelo de calidad de producto que se destaca es: el ISO 25000, que especifica diferentes dimensiones de calidad de producto. El estándar está dividido en cuatro partes las cuales dirigen, realidad, métricas externas, métricas internas y calidad en las métricas de uso y expendido. La calidad de software es un conjunto estructurado de características las cuales son las siguientes:

- a) Funcionalidad: Complejidad, corrección e idoneidad.
- b) Rendimiento: comportamiento en el tiempo y utilización de recursos.
- c) Usabilidad: Inteligibilidad, aprendizaje, operabilidad, protección a errores de usuario, atractividad y accesibilidad.
- d) Fiabilidad: Madurez, disponibilidad, tolerancia a fallos y capacidad de recuperación.
- e) Seguridad: Confidencialidad, integridad, no repudio, autenticidad y responsabilidad.
- f) Mantenibilidad: Modular, reusabilidad, inaplicabilidad, confiabilidad y capacidad de ser probado.
- g) Portabilidad: Adaptabilidad, facilidad de instalación e intercambiabilidad.
- h) Compatibilidad: Coexistencia e interoperabilidad.

Ciclo de vida de la calidad de software específica la calidad de un producto software en un proceso de control al proceso de desarrollo para cada uno de los tipos de calidad como la siguiente figura. (ISO/IEC 25000)

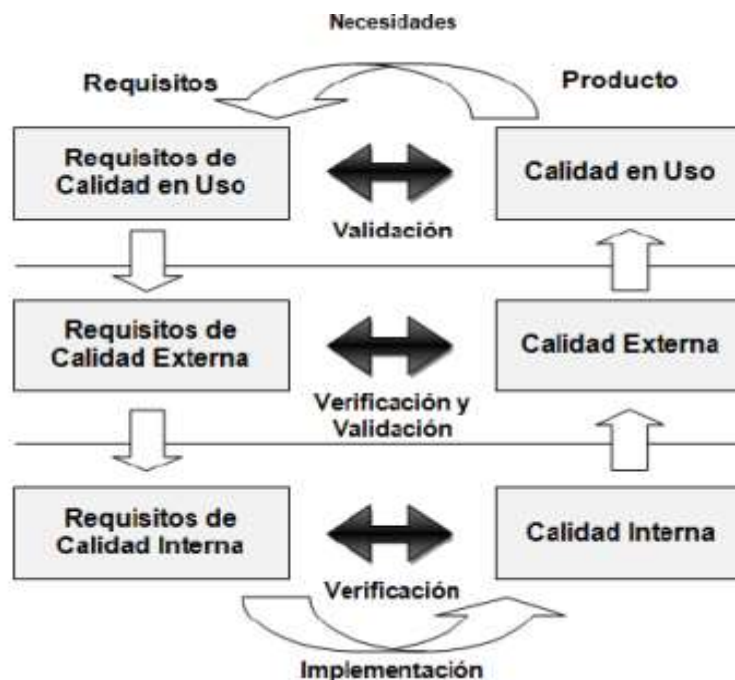


Figura 4. Ciclo de vida de la calidad del producto software. (ISO/IEC 25000)

El ciclo de vida en la calidad del producto software en tres principales fases según ISO/IEC 25000:

- a). Calidad Interna: cuando el producto software se encuentra en desarrollo.
- b). Calidad Externa: cuando el producto software se encuentra en funcionamiento.
- c). Calidad en Uso: cuando el producto software se encuentra en uso.

2.5.3. Modelo de Calidad Interna y Externa

El modelo para la calidad interna y externa de un producto software, define ocho características las cuales a su vez son subdivididos en subcaracterísticas, que pueden ser medidas con métricas internas o externas.



Figura 5. Modelo de calidad interna y externa. (ISO/IEC 25010)

Describimos cada característica para la calidad interna y externa:

- a). Adecuación Funcionalidad: representa la capacidad del producto o sistema software para proporcionar las funciones necesarias para satisfacer al usuario. Esta característica se divide en las siguientes subcaracterísticas:
 - Complejidad funcional: capacidad del sistema software para proporcionar un conjunto de funcionalidades apropiadas para cubrir todas las tareas y objetivos determinados por el usuario. (ISO/IEC 25010, 2005)

- Exactitud funcional: capacidad del sistema software para proporcionar los resultados correctos con el grado necesario de precisión. (ISO/IEC 25010, 2005)

b). Fiabilidad: capacidad del producto o sistema software para realizar las funciones específicas cuando se utiliza bajo ciertas condiciones y periodos de tiempo determinadas. Esta característica se divide en las siguientes subcaracterísticas:

- Madurez: capacidad del sistema software para satisfacer las necesidades de fiabilidad durante el funcionamiento normal. (ISO/IEC 25010, 2005)
- Disponibilidad: capacidad de un sistema software para estar operativo y accesible para su uso cuando se necesite. (ISO/IEC 25010, 2005)
- Tolerancia a Fallos: capacidad de un sistema software para operar cuando se presenten fallos. (ISO/IEC 25010, 2005)
- Recuperabilidad: capacidad de un sistema software para reestablecer el estado del sistema y recuperar datos que se hayan afectado, en caso de interrupción o fallo. (ISO/IEC 25010, 2005)

c). Eficiencia en el Desempeño: capacidad del producto o sistema software de proporcionar un rendimiento apropiado, respecto a la cantidad recursos utilizados bajo determinadas condiciones. Esta característica se divide en las siguientes subcaracterísticas:

- Comportamiento Temporal: capacidad de un sistema software para proporcionar los tiempos de respuesta y procesamiento apropiados. (ISO/IEC 25010, 2005)
- Utilización de Recursos: capacidad en que un sistema software utiliza las cantidades y tipos de recursos adecuados. (ISO/IEC 25010, 2005)
- Capacidad: capacidad de un sistema software de cumplir con los requisitos determinados. (ISO/IEC 25010, 2005)

d). Facilidad de Uso: capacidad del producto o sistema software para que sea entendido, aprendido, agrado y usado por el usuario. Esta característica se divide en las siguientes subcaracterísticas:

- Capacidad de reconocer su adecuación: capacidad del sistema software que permite al usuario entender si el software es adecuado para sus necesidades. (ISO/IEC 25010, 2005)
- Capacidad para ser entendido: capacidad del sistema, que permite al usuario entender si el software es adecuado para alcanzar sus objetivos determinados. (ISO/IEC 25010, 2005)
- Operativo: capacidad de un sistema software que permite al usuario operarlo y controlarlo con facilidad. (ISO/IEC 25010, 2005)
- Protección contra errores del usuario: capacidad en que el sistema brinda la protección necesaria contra errores que realizan los usuarios. (ISO/IEC 25010, 2005)
- Estética de la Interfaz del usuario: capacidad en que la interfaz de usuario llega a satisfacer y agradar al usuario. (ISO/IEC 25010, 2005)
- Accesibilidad técnica: capacidad del sistema software para que se permita ser utilizado por usuarios con determinadas discapacidades. (ISO/IEC 25010, 2005)

e). Seguridad: capacidad para proteger la información y los datos, de manera que personas o sistemas no autorizados puedan tener acceso para consultas o actualizaciones. Esta característica se divide en las siguientes subcaracterísticas:

- Confidencialidad: capacidad de proteger la información y el acceso a datos no autorizados, ya sea de manera accidental o intencional. (ISO/IEC 25010, 2005)
- Integridad: capacidad de un producto, sistema o componente software para evitar accesos no autorizados a datos o programas de computación. (ISO/IEC 25010, 2005)
- No repudio: capacidad para demostrar que los eventos han ocurrido, de manera que dichos eventos no puedan ser refutados posteriormente. (ISO/IEC 25010, 2005)
- Responsabilidad: capacidad de dar seguimiento a las acciones que fueron realizadas por una entidad. (ISO/IEC 25010, 2005)
- Autenticidad: capacidad de demostrar la identidad de un sujeto o un recurso. (ISO/IEC 25010, 2005)

f). Compatibilidad: capacidad de dos o más sistemas, para llevar a cabo sus funciones intercambiando información mientras comparten el mismo entorno. Esta característica se divide en las siguientes subcaracterísticas:

- Co-Existencia: capacidad de un sistema software para coexistir en un entorno en el cual comparten recursos comunes con otro software independiente. (ISO/IEC 25010, 2005)
- Interoperatividad: capacidad de dos o más sistemas software para intercambiar información y utilizar dicha información. (ISO/IEC 25010, 2005)

g). Mantenibilidad: capacidad del sistema software para ser modificado o actualizado debido a necesidades evolutivas o correctivas. Esta característica se divide en las siguientes subcaracterísticas:

- Modularidad: capacidad de un sistema software que cuando sea modificado no afecte a otras funcionalidades del sistema. (ISO/IEC 25010, 2005)
- Reusabilidad: capacidad de un activo (Información, Software, Hardware, Usuarios) para ser utilizado en más de un sistema o en la construcción de otros activos. (ISO/IEC 25010, 2005)
- Capacidad de ser Analizado: facilidad con la que se puede llevar a cabo un análisis del impacto de una determinada modificación en el sistema. (ISO/IEC 25010, 2005)
- Capacidad de ser Modificado: capacidad del sistema para permitir que sea modificado sin causar daños o reducir la calidad del producto. (ISO/IEC 25010, 2005)
- Capacidad de ser probado: facilidad de realizar pruebas a un sistema o componente software, para determinar si se han cumplido con los requerimientos establecidos. (ISO/IEC 25010, 2005)

h). Portabilidad: capacidad de un sistema o componente software de ser trasladado de un entorno a otro sin que esto afecte la funcionalidad de cada sistema. Esta característica se subdivide en las siguientes subcaracterísticas:

- Adaptabilidad: capacidad de un sistema software de ser adaptado a distintos entornos. (ISO/IEC 25010, 2005)

- Capacidad de ser Instalado: capacidad de un sistema para que pueda ser fácilmente instalado y/o desinstalado. (ISO/IEC 25010, 2005)
- Capacidad de ser Reemplazado: capacidad del sistema software para ser utilizado en lugar de otro sistema en el mismo entorno y cumpliendo con el mismo objetivo. (ISO/IEC 25010, 2005)

2.5.4. Modelo para la calidad en uso para el producto software

El modelo de calidad en uso define 5 características, las cuales a su vez son subdivididas en subcaracterísticas, estas son medidas con métricas de calidad en uso.

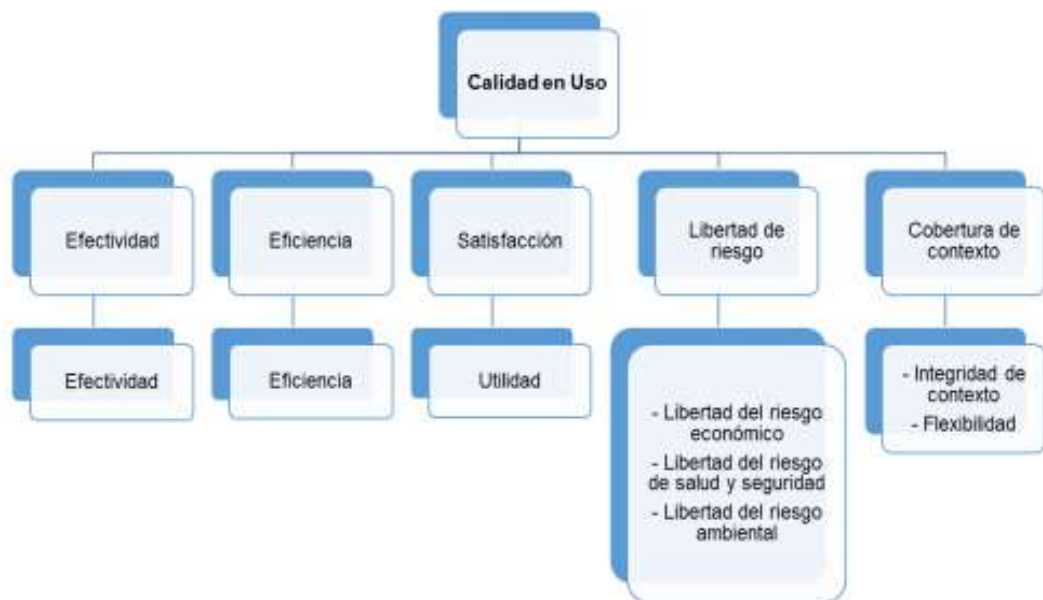


Figura 6. Modelo para la calidad de uso para producto software. (ISO/IEC 25010)

El resultado de calidad en uso depende del logro de la calidad externa, que a su vez depende del logro de la calidad interna. (ISO/IEC 25010, 2005).

Definiremos cada característica y subcaracterísticas para la calidad en uso:

- a). Efectividad: capacidad del sistema software para alcanzar los objetivos o necesidades del usuario, al momento de utilizar el sistema. (ISO/IEC 25000, 2005)
- b). Eficiencia: capacidad del sistema software para alcanzar los objetivos del usuario, utilizando los recursos mínimos. (ISO/IEC 25000, 2005)

c). Satisfacción: capacidad del sistema software para satisfacer las diferentes necesidades mínimas de los usuarios al utilizarlo. Esta característica se divide en las siguiente subcaracterística:

- Utilidad: grado en que un usuario es satisfecho cuando logra alcanzar sus objetivos planteados. (ISO/IEC 25000, 2005)

d). Libertad de Riesgo: capacidad que tiene un producto o sistema software en reducir el riesgo potencial relacionado con la situación económica, vida humana, salud o medio ambiente. (ISO/IEC 25000, 2005)

Esto incluye la salud y seguridad, tanto del usuario y aquellos afectados por el uso, así como las consecuencias materiales o económicas no deseadas. En este caso, el riesgo es la probabilidad de ocurrencia y las posibles consecuencias negativas cuando se presenta una amenaza determinada.

Esta característica se subdivide en las siguientes subcaracterísticas las que permiten establecer el grado en el cual los objetivos podrían estar en riesgo. (ISO/IEC 25000, 2005)

- Libertad del riesgo económico.
- Libertad del riesgo de salud y seguridad.
- Libertad del riesgo ambiental.

e). Cobertura contexto: capacidad de un producto o sistema software para ser utilizado con efectividad, eficiencia, libertad de riesgo y satisfacción en ámbitos de uso que fueron definidos. Esta característica se subdivide en las siguientes subcaracterísticas:

- Integridad de Contexto: capacidad de un sistema software para ser utilizado en los ámbitos de uso definidos. (ISO/IEC 25000, 2005)
- Flexibilidad: capacidad de un sistema software para ser utilizado fuera de los ámbitos de uso definidos inicialmente. (ISO/IEC 25010, 2005)

2.5.5. Prueba de caja negra

Las pruebas de caja negra se definen como una técnica de testing en la que se prueba la funcionalidad de una aplicación ignorando la parte interna de dicha aplicación. Esto quiere decir que se obvia la estructura del código, la arquitectura, los detalles relacionados con la implementación de los diferentes módulos, paquetes o rutas en la que se compone el código. Este

tipo de prueba se basan por completo en los requisitos de las aplicaciones y en sus especificaciones técnicas. En las pruebas de caja negra el Tester solo se centra en las entradas y salidas de la aplicación, sin preocuparse por el contenido interno. Lo que pase por dentro es indiferente, solo importa que, si se realiza cierta acción, la salida sea la indicada según los requerimientos. En las pruebas de caja negra, nos enfocamos solamente en las entradas y salidas del sistema, sin preocuparnos en tener conocimiento de la estructura interna del programa de software. Para obtener el detalle de cuáles deben ser esas entradas y salidas, nos basamos en los requerimientos de software y especificaciones funcionales. (Terrera, 2017)

2.5.6. Prueba de caja blanca

Las pruebas de caja blanca (también conocidas como pruebas de caja de cristal o pruebas estructurales) se centran en los detalles procedimentales del software, por lo que su diseño está fuertemente ligado al código fuente, lo cual, esto significa que tenemos que realizar un “estudio”, o más bien, un análisis de nuestro código, para esto, existen tres pautas fundamentales para poder realizar con éxito una prueba de caja blanca.

Pruebas de cubrimiento: Se trata básicamente de ejecutar al menos una vez cada sentencia. Pero para cumplir con las pruebas de cubrimiento se necesitan varios casos de prueba:

- Determinar posibles “camino” independientes.
- Cada condición debe cumplirse en un caso y en otro no.
- Y puede ser imposible cubrir el 100%
- Código que nunca se ejecuta: condiciones imposibles

Pruebas de condiciones: Cumplir o no cada parte de cada condición. Se necesitan varios casos de prueba:

- Determinar expresiones simples en las condiciones
- Una por cada operador lógico o comparación
- Cada expresión simple debe cumplirse en un caso y en otro no, siendo decisiva en el resultado.

Es por ello que se considera a la prueba de Caja Blanca como uno de los tipos de pruebas más importantes que se le aplican al software, logrando como resultado que disminuya en un gran porcentaje el número de errores

existentes en los sistemas y por ende una mayor calidad y confiabilidad. (Terrera, 2017)

2.5.7. Cobertura de código

Las pruebas unitarias de cobertura de código es un factor importante para determinar qué tan buena es nuestra calidad de software. La cobertura de código lo que nos dice es la cantidad de código que está sometido a nuestras pruebas. A mayor cobertura mayor cantidad de código está siendo probado por nuestras pruebas unitarias. Una cobertura del 85-90% indica que la gran mayoría de nuestro código estaría siendo probado. Una cobertura menor indica que hay una parte importante de nuestra aplicación que está sin probar y que deberíamos completar nuestras para cubrir los escenarios que no están siendo probados. Desde la configuración de las pruebas unitarias debemos establecer los ensamblados que deben instrumentalizar para comprobar la cobertura de código. Por ello la cobertura de código garantiza que mediante el 85 y 90% el software tiene una alta aprobación para su implementación. Los diferentes escenarios que se presentan y plantean permitirán garantizar las pruebas de calidad de software. (Landa, 2010)

CAPÍTULO III. INGENIERÍA DEL PRODUCTO O DE DESARROLLO DE LA INVESTIGACIÓN

3.1. Etapa de inicio

3.1.1. Descripción de la empresa

a) Antecedentes

La Institución educativa Colegio Matemático Honores del Zapallal se localiza en el distrito de Puente Piedra, provincia de Lima, esta institución es supervisada por la UGEL 04 COMAS y esta última pertenece a la Gerencia regional de educación DRE LIMA METROPOLITANA. En la institución educativa Colegio Matemático Honores De Zapallal cuya misión es impartir educación y perfeccionar sus habilidades personales con un aumento en su autoestima, inteligencia, competencias sociales y una consistente educación académica, moral y emocional para que puedan obtener el éxito personal y profesional en una sociedad cambiante. El objetivo de esta institución de Puente Piedra es suministrar una educación de la más alta calidad, en donde los estudiantes alcancen su total crecimiento espiritual, físico, social, intelectual, moral y emocional.

b) Dirección

MZ B LOTE 5, Puente Piedra

c) RUC

20522983421

d) Apoderado principal

Alejandro Dagoberto Agama Fernández, promotor y director del Colegio Matemático Honores del Zapallal.

e) Organigrama

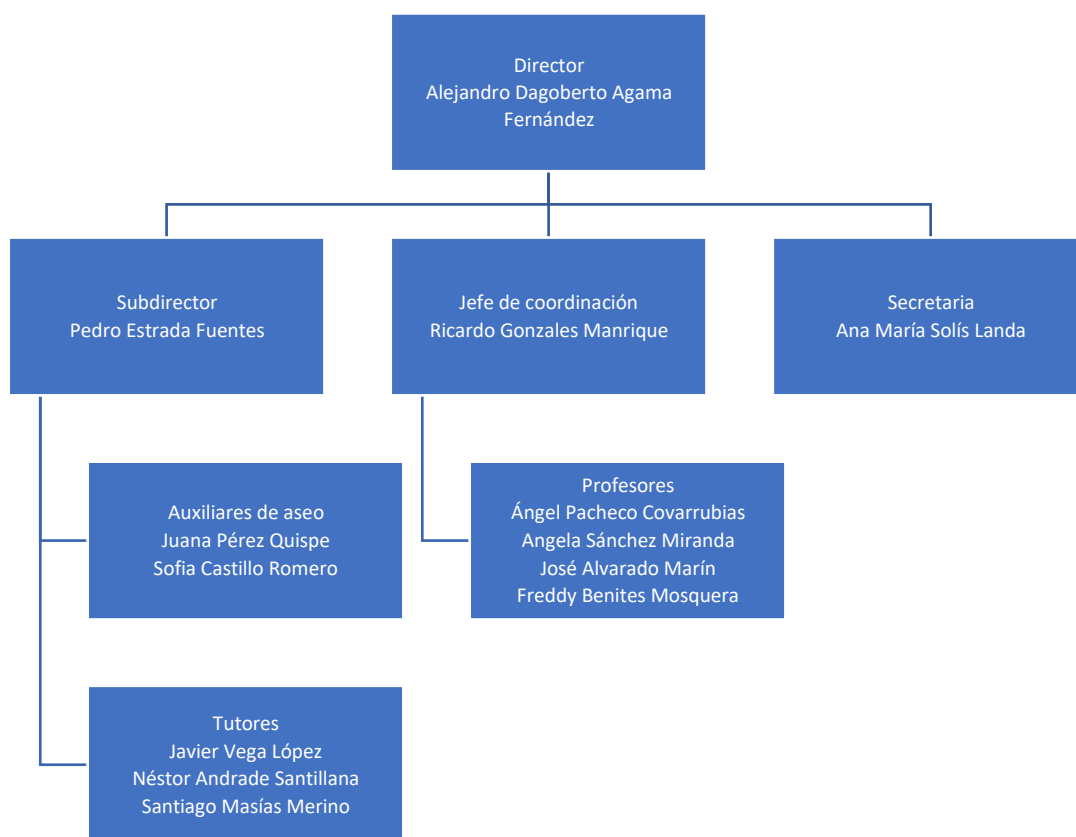


Figura 7. Organigrama del Colegio Matemático Honores de Zapallal

3.1.2. Descripción global del software

a) Requerimientos

Tabla 2. Requisitos del Sistema.

N.º	Requisito	Tipo de Requisito
01	El sistema deberá manejar un login para usuarios.	funcional
02	El sistema deberá manejar roles de usuario.	funcional
03	El sistema deberá registrar los datos del apoderado	funcional
04	El sistema deberá manejar los privilegios de los usuarios.	funcional
05	El sistema deberá registrar los datos de los estudiantes.	funcional
06	El sistema deberá consultar los datos de los estudiantes	funcional
07	El sistema deberá consultar los datos de los apoderados	funcional
08	El sistema deberá registrar los datos de parentesco	funcional
09	El sistema deberá consultar los datos de parentesco	funcional
10	El sistema deberá registrar los datos de la matricula	funcional
11	El sistema deberá actualizar los datos de la matricula	funcional

12	El sistema deberá consultar el recaudo	funcional
13	El sistema deberá registrar los datos de los docentes	funcional
14	El sistema deberá consultar los datos del docente	funcional
15	El sistema deberá actualizar los datos del docente	funcional
16	El sistema deberá consultar el horario del docente	funcional
17	El sistema deberá registrar la calificación	funcional
18	El sistema deberá consultar la calificación	funcional
19	El sistema deberá actualizar la calificación	funcional
20	El sistema deberá registrar el año escolar	funcional

b) Casos de uso

Tabla 3. Casos de uso.

N.º	Requerimiento	CUS	Descripción del CUS
01	El sistema deberá manejar un login para usuarios.	Seguridad	En este CUS, se validará el ingreso al sistema a través de un Login
02	El sistema deberá manejar roles de usuario.	Gestión de Usuarios	En este CUS el administrador registrará, modificará y listará los usuarios.
03	El sistema deberá manejar los privilegios de los usuarios.		
04	El sistema deberá registrar los datos del apoderado.	Gestión de Datos	En este CUS la secretaria y los profesores registrarán, consultarán y actualizarán los datos según su acceso.
05	El sistema deberá registrar los datos de los estudiantes.		
06	El sistema deberá consultar los datos de los estudiantes		
07	El sistema deberá consultar los datos de los apoderados		
08	El sistema deberá registrar los datos de parentesco		
09	El sistema deberá consultar los datos de parentesco		
10	El sistema deberá registrar los datos de la matrícula		
11	El sistema deberá actualizar los datos de la matrícula		
12	El sistema deberá consultar el recaudo		

13	El sistema deberá registrar los datos de los docentes		
14	El sistema deberá consultar los datos del docente		
15	El sistema deberá actualizar los datos del docente		
16	El sistema deberá consultar el horario del docente		
17	El sistema deberá registrar la calificación		
18	El sistema deberá consultar la calificación		
19	El sistema deberá actualizar la calificación		
20	El sistema deberá registrar el año escolar		

c) Diagrama de casos de uso

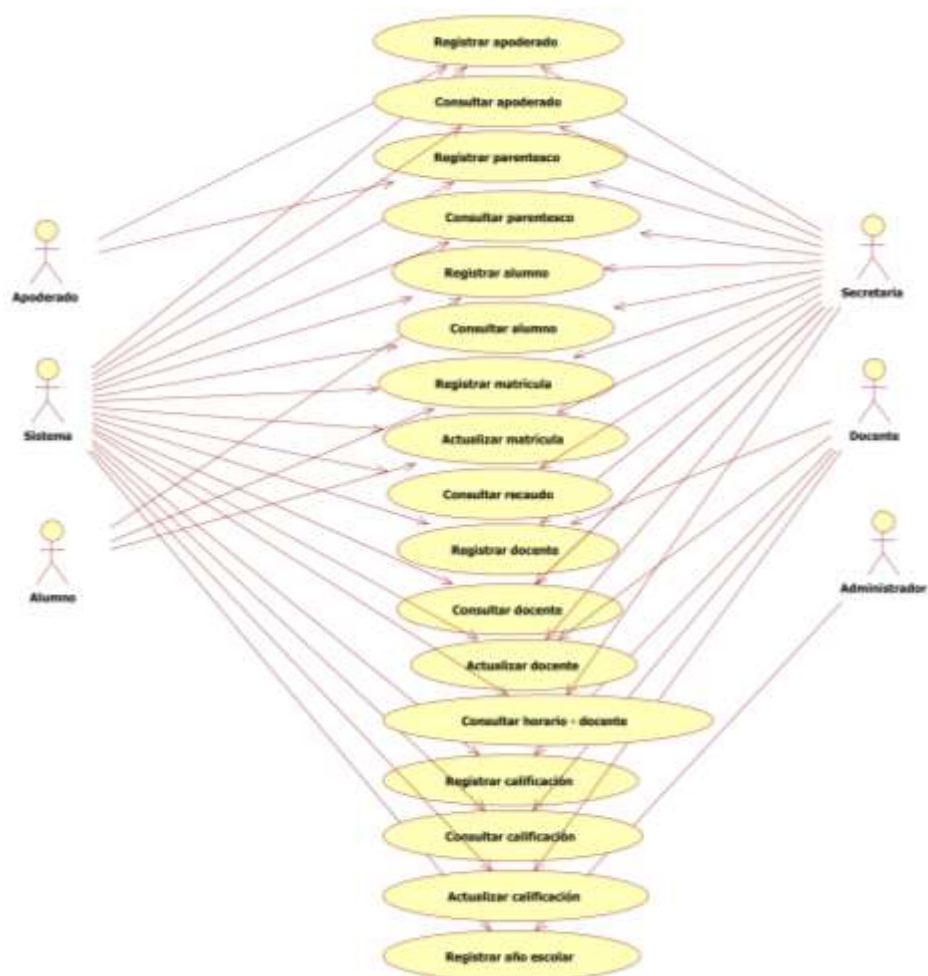


Figura 8. Diagrama de uso.

d) Arquitectura del software y de hardware

- Arquitectura de Software:

Se uso la arquitectura cliente-servidor llamado programación por capas con el objetivo de separar la lógica de negocio, diseño de interfaz y el acceso de datos. La capa de presentación es la que el usuario visualizará, este presenta el sistema al usuario, le brinda y registra la información, además de que esta capa es la única que se comunica con la capa de negocio. La capa de negocio es donde se alberga la lógica del negocio y es donde el usuario realiza sus peticiones y en consecuencia se envía las respuestas, además de que esta capa es la única que se comunica con la capa de presentación y la capa de datos. Al comunicarse con la capa de datos solicita almacenar o recuperar datos. La capa de datos es la encargada de conectarse a la base de datos por lo cual se encarga de recibir las acciones de la capa lógica de negocio.



Figura 9. Programación por Capas.

- Arquitectura de Hardware:



Figura 10. Arquitectura de Hardware

a) Principios de ingeniería de software

Figura 11. Principios de Ingeniería de Software

b) Principios de gestión de proyectos

[illegible]

Figura 12. Principios de Gestión de Proyectos

c) Principios de código fuente

DOMINIO	SUB-DOMINIO	ETIQUETA	ID	PREGUNTA	ARTIFACTO	SI CUMPLE	NO CUMPLE	REVISOR	COMENTARIO	EVIDENCIA	VALOR CALIDAD POR PREGUNTA	NÚMERO CALIDAD SUB-DOMINIO	NÚMERO CALIDAD DOMINIO
ESTÁNDARES DE CODIFICACIÓN			1.1.1	1	Lineas de comentario por funcion	Codigo fuente	X		Max Páucar	El comentario aparece en el código fuente	EV-001	5	100%
			1.1.2	2	Lineas de comentario por modulo	Codigo fuente	X		Adonal Huaraz	El comentario aparece en el código fuente	EV-002 (Aprendizado), EV-003 (Aulas), EV-004 (Notas), EV-005 (Matriculas), EV-006 (Docentes), EV-007 (Año escolar)	5	
			1.1.3	3	Inicialización de variables	Codigo fuente	X		Max Páucar	La inicialización de variables aparece en el código fuente	EV-008	5	
			1.1.4	4	¿Todas las variables del programa se inician antes de usar sus valores?	Codigo fuente	X		Adonal Huaraz	Todas las variables se inician antes de tener valor, como se muestra en el github	EV-012	5	
			1.1.5	5	¿Todas las constantes tienen nombre?	Codigo fuente	X		Max Páucar	No se requiere el uso de constantes	EV-013	1	
			1.1.6	6	¿La cota superior de los arreglos es igual al tamaño del arreglo o valor = 1?	Codigo fuente	X		Adonal Huaraz	Implementados como muestra la imagen en el github	EV-014	5	
			1.1.7	7	Si se usan cadenas de caracteres, ¿Se asigna explícitamente un delimitador?	Codigo fuente	X		Max Páucar	No se requiere el uso de delimitadores		1	
			1.1.8	8	¿Existe alguna posibilidad de desbordamiento de buffer?	Codigo fuente	X		Adonal Huaraz	No existe porque se está usando el metodo clear() para borrar la memoria asignada	EV-015	1	
			1.1.9	9	¿Para cada enunciado condicional, ¿La condición es correcta?	Codigo fuente	X		Max Páucar	Se uso de forma efectiva las sentencias condicionales	EV-016	5	
			1.1.10	10	¿Hay certeza de que termine cada ciclo?	Codigo fuente	X		Adonal Huaraz	Los ciclos del código fuente cumplen su función	EV-017	5	
			1.1.11	11	¿Los enunciados compuestos están correctamente colocados entre paréntesis?	Codigo fuente	X		Max Páucar	Se ha respetado la estructura de los códigos compuestos	EV-018	5	
			1.1.12	12	En caso de enunciados, ¿se justifican todos los casos posibles?	Codigo fuente	X		Adonal Huaraz	Se cumple con la especificación requerida	EV-019	5	
			1.1.13	13	Si después de cada caso en los enunciados se requiere un paréntesis, ¿este se incluye?	Codigo fuente	X		Max Páucar	Cada enunciado cuenta con su paréntesis respectivo	EV-020	5	
			1.1.14	14	¿Se usan todas las variables de entrada?	Codigo fuente	X		Adonal Huaraz	Se cumple con la especificación requerida	EV-021	5	
FALLAS DE ENTRADA / SALIDA			1.1.15	15	¿A todas las variables de salida se les asigna un valor antes de que se produzcan?	Codigo fuente	X		Max Páucar	Se cumple con la especificación requerida	EV-022	5	100%
			1.1.16	16	¿Entradas inesperadas pueden causar corrupción?	Codigo fuente	X		Adonal Huaraz	Se cumple con la especificación requerida	EV-026	5	
			1.1.17	17	¿Se usan todas las variables de entrada?	Codigo fuente	X		Max Páucar	Se cumple con la especificación requerida	EV-024	5	
FALLAS DE INTERFAZ			1.1.18	18	¿A todas las variables de salida se les asigna un valor antes de que se produzcan?	Codigo fuente	X		Adonal Huaraz	Se cumple con la especificación requerida	EV-025	5	100%
			1.1.19	19	¿Entradas inesperadas pueden causar corrupción?	Codigo fuente	X		Max Páucar	Los ingresos de datos estan validados	EV-023	5	
			1.1.20	20	¿Todas las llamadas a funcion y metodo tienen el numero correcto de parametros?	Codigo fuente	X		Adonal Huaraz	Se cumple la cantidad de parametros requerida por funcion	EV-028	5	
FALLAS DE GESTIÓN DE ALMACENAMIENTO			1.1.21	21	¿Los tipos de parametros formal y real coinciden?	Codigo fuente	X		Max Páucar	Si hay coincidencia entre los dos parametros	EV-027	5	100%
			1.1.22	22	¿Los parametros estan en el orden correcto?	Codigo fuente	X		Adonal Huaraz	Si se cumple con la especificación requerida	EV-029	5	
			1.1.23	23	Si los componentes acceden a memoria compartida, ¿Tienen el mismo modelo de estructura de memoria compartida?	Codigo fuente	X		Max Páucar	Los componentes al hacer la llamada se cargan en un DataGridView	EV-030	5	
			1.1.24	24	Si se modifica una estructura vinculada, ¿Todos los vinculos se reasignan correctamente?	Codigo fuente	X		Adonal Huaraz			5	
			1.1.25	25	Si se usa almacenamiento dinamico, ¿el espacio se asigna correctamente?	Codigo fuente	X		Max Páucar	No se requiero usar almacenamiento dinamico	EV-031	1	
FALLAS DE GESTIÓN DE EJECUCIÓN			1.1.26	26	¿El espacio se libera explícitamente después de que ya no se requiere?	Codigo fuente	X		Adonal Huaraz	Se utilizó el metodo Trim para cancelar los espacios en blanco	EV-032	5	80%
			1.1.27	27	¿Se tomaron en cuenta todas las posibles condiciones de error?	Codigo fuente	X		Max Páucar	Se tomaron las excepciones y las validaciones para los posibles errores	EV-033	5	
PATRON MVC			1.1.28	28	Clase de software esta codificada para Vista Clase de software esta codificada para control Clase de software esta codificada para modelo?	Codigo fuente	X	Adonal Huaraz	No cumple con las caracteristicas del patron MVC	EV-009	1	20%	
PATRON ARQUITECTURA			1.1.29	29	Los procedimientos SQL se ejecutan solo en procedimientos almacenados	Procedimientos almacenados en BD	X	Max Páucar	Los procedimientos almacenados se encuentran dentro de la base de datos	EV-010	5	100%	
			1.1.30	30	Script de creacion BD	Scripts en BD	X	Adonal Huaraz	La creacion de la BD se encuentra en el Query	EV-011	5		

Figura 13. Principios de Código Fuente

d) Desarrollo del modelo GQM aplicado al proyecto.

DIRECCION DE PROCESO					
DOMINIO	META	PREGUNTAS	METRICA	FRECUENCIA	ARTEFACTO (FUENTE)
INGENIERIA DE SOFTWARE	Tener totalmente especificados los casos de uso	¿Cómo asegurar que estos totalmente especificados los casos de uso?	N° casos de uso especificados/N° casos de uso establecidos	Semanal	Cuadro de especificaciones
	Tener totalmente identificados los actores del negocio	¿Cómo se asegura que estos totalmente identificados los actores del negocio?	N° actores del negocio/N° actores del negocio establecidos	Semanal	Cuadro de actores del negocio
	Tener totalmente identificados los casos de uso del negocio	¿Cómo se asegura que estos totalmente identificados los casos de uso del negocio?	N° casos de uso del negocio/N° casos de uso del negocio establecidos	Semanal	Cuadro de casos de uso del negocio
	Realizar el diagrama de casos de uso del negocio con los casos de uso	¿Cómo se asegura que todos los casos de uso del negocio tengan un diagrama de casos de uso?	Diagrama de casos de uso del negocio/procesos definidos en cada CU	Semanal	Diagrama de casos de uso del negocio
	Tener totalmente identificados los casos de uso del sistema	¿Cómo se asegura que estos totalmente identificados los casos de uso del sistema?	N° casos de uso del sistema/N° casos de uso del sistema establecidos	Semanal	Cuadro de casos de uso del sistema
	Realizar los diagrama de actividades por procesos de los casos de uso	¿Cómo asegurar que todos los procesos de los casos de uso tengan su respectivo diagrama de actividades?	N° diagramas de actividades en un CU/N° procesos definidos en un CU	Semanal	Diagrama de actividades
	Utilizar una arquitectura de software para el sistema	¿Cómo saber si la implementación de la arquitectura ha sido desarrollada?	N° hojas de código desarrollados/N° hojas de código definidos	Semanal	Descripción del proyecto
	Utilizar una guía de usuario para el sistema	¿Cómo saber si la guía de usuario ha sido desarrollada?	N° formularios desarrollados/N° formularios definidos	Semanal	Guía de usuario
	Utilizar una metodología para la implementación del proyecto	¿Cómo saber si la metodología ha sido desarrollada?	N° entregables desarrollados/N° entregables definidos	Semanal	Documentación del proyecto
	Desarrollar todos los requisitos definidos por módulos	¿Cómo saber si los requisitos definidos por módulos han sido desarrollados?	N° requisitos desarrollados/N° requisitos definidos	Ad-hoc	Lista de requisitos del sistema
GESTION DE PROYECTO	Entregar los documentos completos de la constitución del proyecto	¿Cómo entregar los acuerdos establecidos de la constitución del proyecto?	N° acuerdos cumplidos/N° acuerdos establecidos	Semanal	Acta de constitución - Informe del proyecto
	Realizar el cumplimiento de todas las estipulaciones del acta de constitución	¿Cómo se asegura la identificación de todas las estipulaciones después del cierre del proyecto?	N° de estipulaciones después de cierre del proyecto/N° de estipulaciones definidas después del cierre del proyecto	Semanal	Acta de constitución
	Entregar el cronograma del proyecto	¿Cómo se definen las actividades del cronograma del proyecto?	N° acuerdos cumplidos/N° acuerdos establecidos	Semanal	Cronograma del proyecto - Gantt
	Mejorar la planificación de la gestión del proyecto	¿Cómo realizar la mejora de planificación de la gestión del proyecto?	N° documentos cumplidos/N° documentos establecidos	Semanal	Qui, EOI - Cronograma del proyecto - Plan de presupuesto - Plan de gestión
	Mejorar los procesos ejecución del proyecto	¿Cómo realizar la mejora de los procesos de ejecución del proyecto?	N° procesos ejecutados cumplidos/N° procesos ejecución establecidos	Semanal	Plan de ejecuciones - Plan de gestión entregables - Plan de gestión de cambios
	Mejorar el monitoreo y control del proyecto	¿Cómo realizar la mejora del monitoreo y control del proyecto?	N° procesos monitoreados/N° procesos control establecidos	Semanal	Informe de monitoreo y control - Informe de gestión de calidad - Informe de avances y cronogramas
	Garantizar la conformidad del cierre total del proyecto	¿Cómo garantizar la conformidad del cierre total del proyecto?	N° conformidad aceptada/N° conformidad establecidos	Semanal	Qui, Cierre, legal y administrativo - Qui, Incidencia - Informe Final
	Garantizar la identificación de los riesgos	¿Cómo se realiza la identificación de los riesgos?	N° de riesgos aceptados/N° riesgos establecidos	Semanal	Plan de gestión de riesgos
	Garantizar la conformidad del análisis cualitativo de riesgos	¿Cómo se realiza el análisis cualitativo de riesgos?	N° cualitativos de riesgos aceptados/N° cualitativo de riesgos establecidos	Semanal	Plan de gestión de riesgos
	Garantizar la conformidad del análisis cuantitativo de riesgos	¿Cómo se realiza el análisis cuantitativo de riesgos?	N° cuantitativos riesgos aceptados/N° cuantitativos riesgos establecidos	Semanal	Plan de gestión de riesgos
PROGRAMACIÓN	Garantizar la conformidad del plan de comunicaciones	¿Cómo se realiza el plan de las comunicaciones?	N° de tipos de comunicaciones realizadas/acepta/N° de tipos de comunicaciones establecidos	Semanal	Plan de gestión de comunicaciones
	Tener los sistemas como los variables con un error del defecto	¿Todos los variables del programa se tienen antes de usar los sistemas?	N° variables del programa/N° variables establecidos	Semanal	Código fuente de programas
	Tener el mayor control de los fallos del software	¿Cómo tener el mayor control de los fallos del software?	N° errores con fallos/N° de errores establecidos	Semanal	Código fuente de programas
	Tener el mayor control de los fallos de entrada	¿Cómo tener el mayor control de los fallos de entrada del software?	N° errores con fallos/N° errores	Semanal	Código fuente de variables de entrada
	Tener el mayor control de los fallos de salida	¿Cómo tener el mayor control de los fallos de salida del software?	N° errores con fallos/N° salidas establecidos	Semanal	Código fuente de variables de salida
	Tener correctamente todas las funciones	¿Todas las funciones del programa se conservan?	N° conservadas del programa/N° de conservadas programadas	Semanal	Código fuente de líneas de programación por función
	Tener todos los procedimientos correctos	¿Todos los procedimientos establecidos según condiciones correctas?	N° errores con procedimientos del programa/N° errores establecidos	Semanal	Código fuente de variables condicionales
	Tener todos los procedimientos de programación se preservan	¿Todos los procedimientos de programación del programa se preservan?	N° N° del programa/N° SP aceptadas	Semanal	Quota de SP
	Tener el mayor control de los fallos de errores	¿Cómo tener el mayor control de los fallos de errores del software?	N° fallos en la programación/compensados en la errores	Semanal	Código fuente de fallos de errores

Figura 14. Modelo GQM

a) Calidad del proceso de ingeniería de software

[illegible]

Figura 15. Calidad del Proceso de Ingeniería de Software

a.1) Modelo

1.1.1 ¿Se evaluó el objetivo de la organización?



Acta-de-Constitución.docx

1.1.2 ¿Se identificaron los actores del negocio?



Actor	Descripción de tareas
 Apoderado	Persona que solicita el servicio de matrícula en la organización.
 Alumno	Persona que solicita el servicio de matrícula en la organización.

Figura 16. Caso de uso

1.1.3 ¿Se identificaron los casos de uso del negocio?


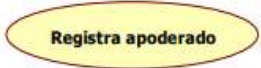

Casos de uso	Descripción de tareas
 Pagos	Caso de uso donde el apoderado realiza los pagos por el servicio brindado.
 Consultas	Caso de Uso encargado de conocer los servicios que brinda el centro educativo.
 Registra apoderado	Caso de uso encargado de registrar los apoderados de los alumnos matricularlos.
 Registra parentesco	Caso de uso encargado de registrar a el parentesco entre apoderados y alumnos matricularlos.
 Registrar alumno	Caso de uso encargado de registrar a los alumnos para matricularlos.
 Matricula	Caso de uso encargado de la gestión de matrículas de los alumnos, de todos los niveles.

Figura 17. Caso de uso de negocio.

1.1.4 ¿Existe diagrama de casos de uso del negocio?

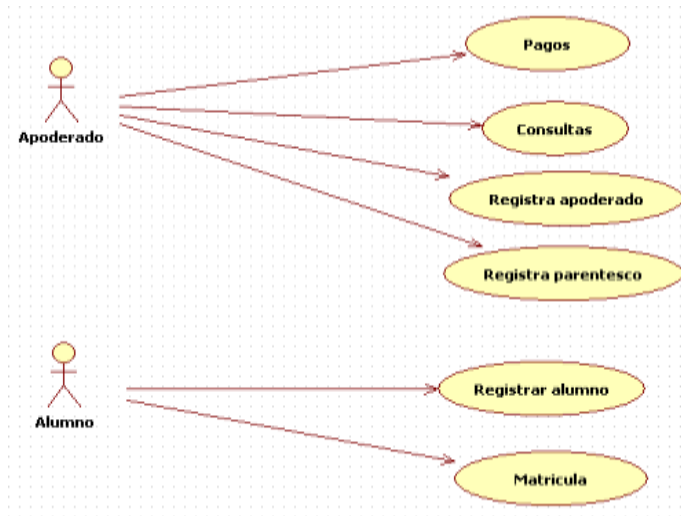


Figura 18. Diagrama de casos de uso

1.1.5 Existe las especificaciones de casos de uso

Caso de uso:	Registrar parentesco
Actores:	Secretaria, apoderado
Propósito:	Conocer la consanguinidad de los parientes de los estudiantes
Resumen:	Existirá un formulario de registro de parentesco, la cual contiene las casillas para ingresar el número de DNI del apoderado y el número de DNI del alumno, se podrá seleccionar el parentesco desde un select, seguidamente se presiona el botón "registrar" y si los números de Dni tanto del apoderado o del alumno no están registrados previamente no se podrá registrar en el sistema.
Tipo:	Alta

Actor	Sistema
Se activa la casilla de verificación "iniciar"	Se habilita las casillas DNI alumno, DNI apoderado y el select de parentesco
Ingresa DNI del alumno, el DNI del apoderado y selecciona el parentesco; y posteriormente se presiona el botón registrar	Realiza la comprobación necesaria en la base de datos si son datos no válidos se informa a la persona, en caso de que todo esté bien, el usuario puede registrar el parentesco en el sistema.

Figura 16. Especificaciones de caso de uso.

a.2) Requerimiento

1.1.6 ¿Se identificaron los actores del sistema?




Actor	Descripción de tareas:
 Administrador	Administración desde la perspectiva del sistema
 Secretaria	Gestión del sistema
 Docente	Enseñanza pedagógica

Figura 19. Actores del sistema.

1.1.7 ¿Se identificaron los casos de uso del sistema?

Casos de uso	Descripción de tareas
Registrar parentesco	Caso de Uso encargado de registrar el parentesco en el sistema.
Registrar apoderado	Caso de Uso encargado de registrar apoderado en el sistema.
Registrar matricula	Caso de Uso encargado de registrar la matricula en el sistema.
Registrar docente	Caso de Uso encargado de registrar el docente en el sistema.
Registrar alumno	Caso de Uso encargado de registrar el alumno en el sistema.
Registrar año escolar	Caso de Uso encargado de registrar el año escolar en el sistema.
Registrar calificación	Caso de Uso encargado de registrar la calificación en el sistema.
Consultar apoderado	Caso de Uso encargado de consultar el apoderado en el sistema.
Consultar parentesco	Caso de Uso encargado de consultar el parentesco en el sistema.

Figura 20. Casos de uso del sistema.

1.1.8 ¿Construcción del diagrama de casos de uso del sistema?

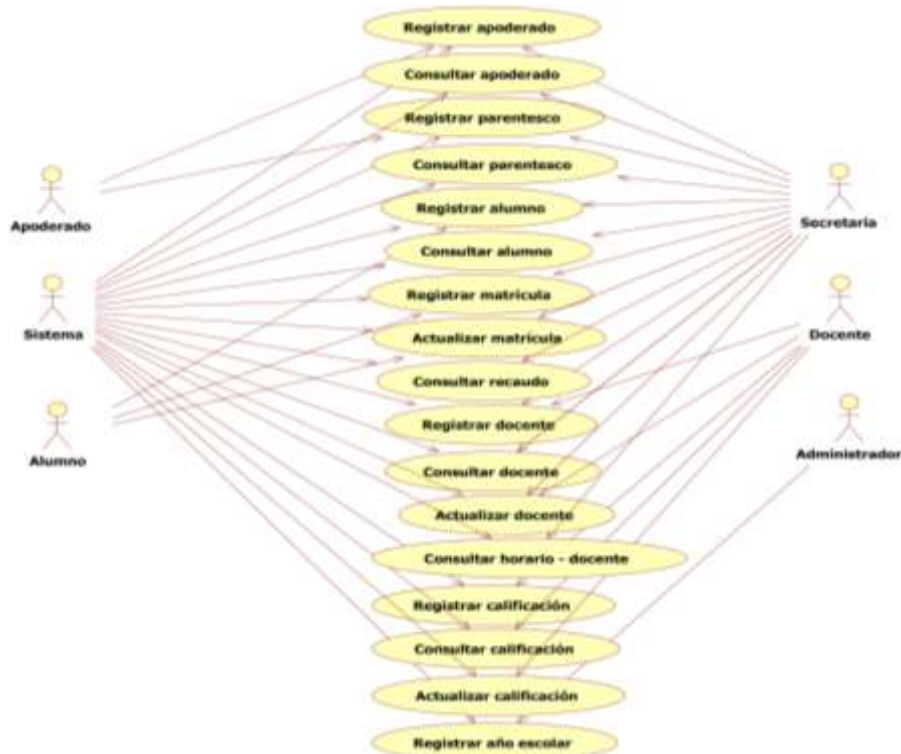


Figura 21. Diagramas de caso de uso de sistema.

1.1.9 ¿Análisis de requerimiento funcionales?

N.º	Requisito	Tipo de Requisito
01	El sistema deberá manejar un login para usuarios.	funcional
02	El sistema deberá manejar roles de usuario.	funcional
03	El sistema deberá registrar los datos del apoderado	funcional
04	El sistema deberá manejar los privilegios de los usuarios.	funcional
05	El sistema deberá registrar los datos de los estudiantes.	funcional
06	El sistema deberá consultar los datos de los estudiantes	funcional
07	El sistema deberá consultar los datos de los apoderados	funcional
08	El sistema deberá registrar los datos de parentesco	funcional
09	El sistema deberá consultar los datos de parentesco	funcional
10	El sistema deberá registrar los datos de la matricula	funcional
11	El sistema deberá actualizar los datos de la matricula	funcional
12	El sistema deberá consultar el recaudo	funcional
13	El sistema deberá registrar los datos de los docentes	funcional
14	El sistema deberá consultar los datos del docente	funcional
15	El sistema deberá actualizar los datos del docente	funcional
16	El sistema deberá consultar el horario del docente	funcional
17	El sistema deberá registrar la calificación	funcional
18	El sistema deberá consultar la calificación	funcional
19	El sistema deberá actualizar la calificación	funcional
20	El sistema deberá registrar el año escolar	funcional

Figura 22. Requerimientos funcionales del sistema.

1.1.10 ¿Análisis de requerimientos no funcionales?

N.º	Requisito	Tipo de Requisito
01	El sistema maneja mensajes de confirmaciones.	No funcional
02	El sistema debe permitir futuras mejoras de acuerdo a las necesidades presentadas.	No funcional
03	La interfaz gráfica se ha creado de una manera de fácil comprensión para el usuario de manera que este no requiera mayor esfuerzo para utilizar el sistema.	No funcional
04	El sistema debe garantizar seguridad, se requiere identificación y contraseña del usuario para acceder al sistema.	No funcional

Figura 23. Requerimientos no funcionales del sistema.

1.1.11 ¿Análisis del requerimiento de implementación?

Requisitos	
Procesador	Intel Core2Duo 3.00 GHZ o superior
Espacio en disco	1 TB mínimo
Memoria RAM	4 GB de DDR3 o superior
Servidor	Sistema operativo Linux
Espacio en servidor	4 TB mínimo

Figura 24. Requerimientos para la implementación del sistema.

1.1.12 Existe el diagrama de actividades

SISTEMA DE COLEGIO

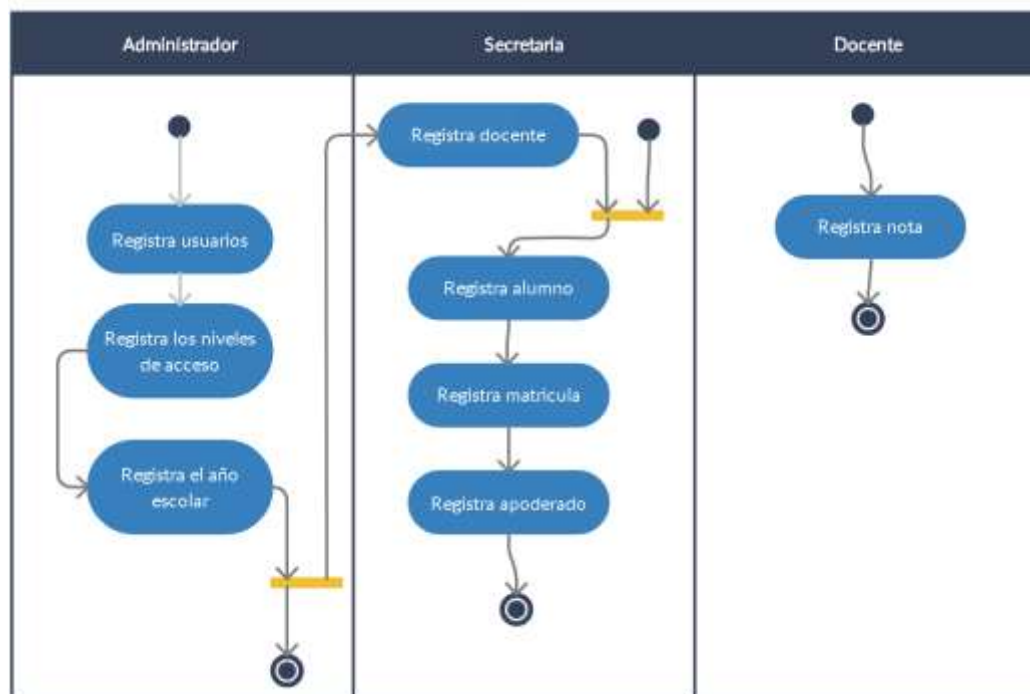


Figura 25. Diagrama de actividades.

a.3) Análisis y diseño

1.1.13 ¿Módulos correspondientes?

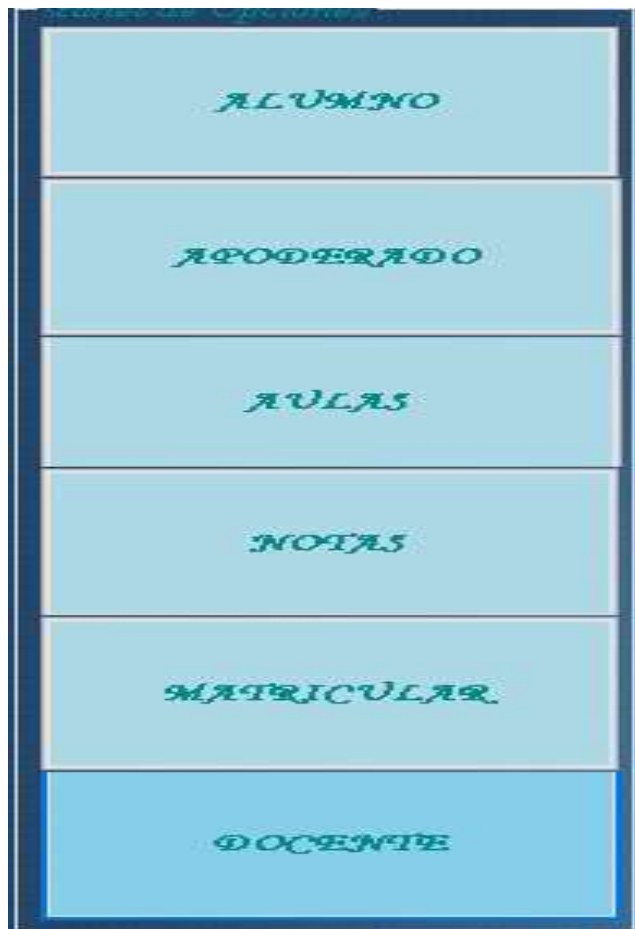


Figura 26. Diseño de módulos del sistema.

1.1.14 ¿Prototipos del diseño?

No existe el prototipo, ya que no hubo necesidad de realizarlo.

1.1.15 ¿Framework necesarios?

Requisitos
Sistema Operativo Windows 10 Pro
Lenguaje de Programación Visual Basic
Microsoft SQL Server 19
Framework Visual Studio 19.
Microsoft SQL Server Management Studio (SSMS).
Microsoft Office 2019

Figura 27. Marcos de trabajo necesarios para la implementación del sistema.

1.1.17 Existe el diagrama de colaboración



1.1.18 Existe el diagrama de secuencia

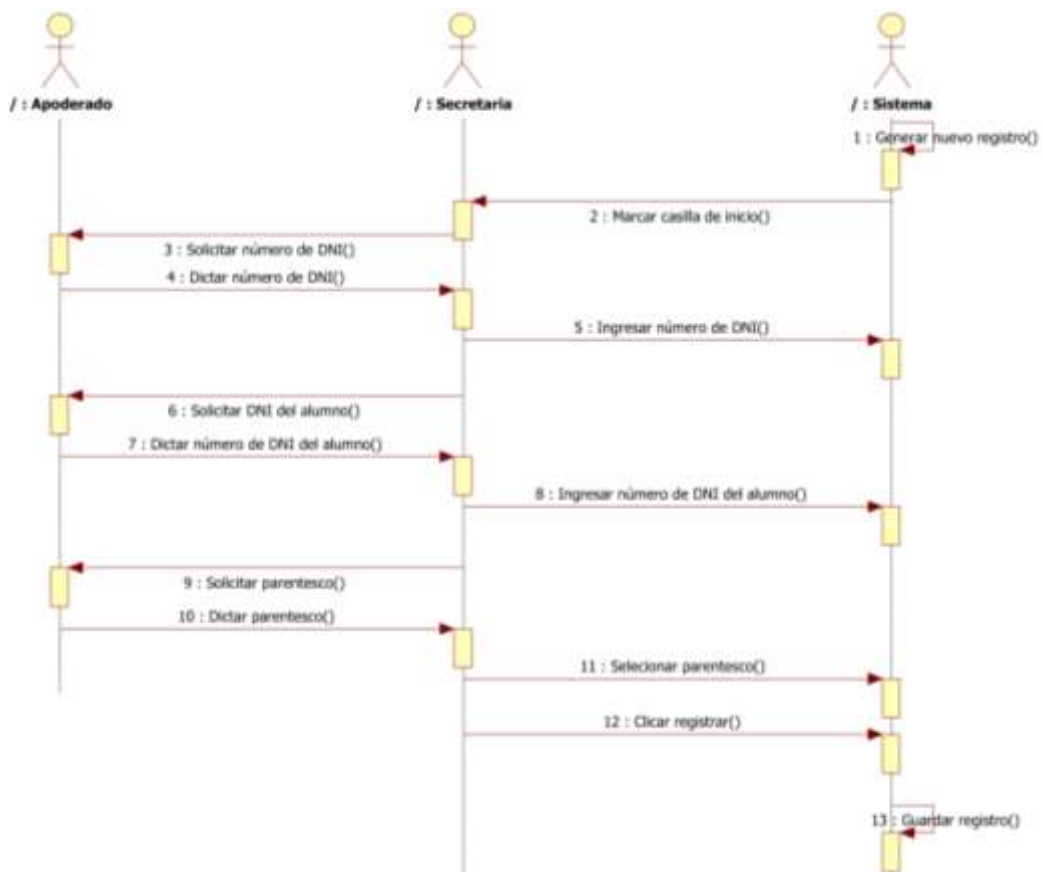


Figura 30. Diagrama de secuencias.

1.1.19 Existe el diagrama de clases

No existe el diagrama de clases.

1.1.20 Existe el modelo de la base de datos

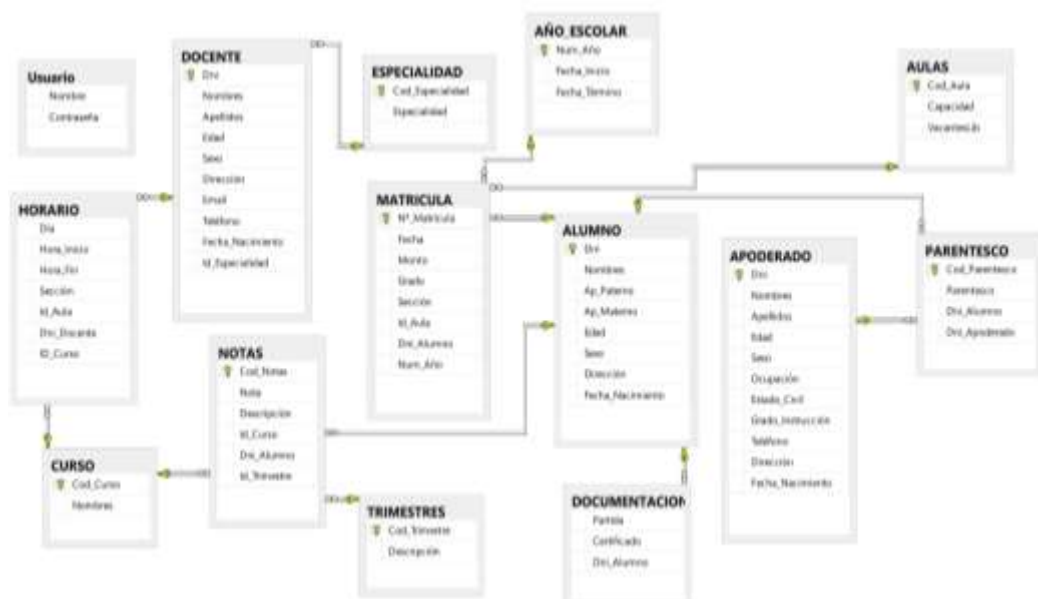
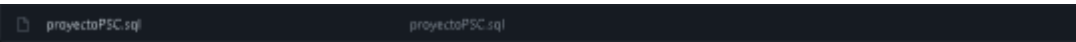


Figura 31. Modelo de la base de datos.

1.1.21 ¿Creación de la base de datos?



1.1.22 ¿Programación de interfaces?

```
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()> _
Partial Class Inicio
    Inherits System.Windows.Forms.Form

    'Form provides a Dispose method to dispose the list of components.
    <System.Diagnostics.DebuggerNonUserCode()> _
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        Try
            If disposing AndAlso components IsNot Nothing Then
                components.Dispose()
            End If
        Finally
            MyBase.Dispose(disposing)
        End Try
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer


    'NOTE: The Windows Form Designer needs the following procedure
    'to be able to add components to the Windows Form.
    'Do not modify this code.
    <System.Diagnostics.DebuggerStepThrough()> _
    Private Sub InitializeComponent()
        Me.components = New System.ComponentModel.Container()
        Dim resources As System.ComponentModel.ComponentResourceManager = New System.ComponentModel.ComponentResourceManager(GetType(Inicio))
        Me.Label1 = New System.Windows.Forms.Label()
        Me.ProgressBar1 = New System.Windows.Forms.ProgressBar()
        Me.Timer1 = New System.Windows.Forms.Timer(Me.components)
        Me.PictureBox1 = New System.Windows.Forms.PictureBox()
        Me.PictureBox2 = New System.Windows.Forms.PictureBox()
        CType(Me.PictureBox1, System.ComponentModel.ISupportInitialize).BeginInit()
        CType(Me.PictureBox2, System.ComponentModel.ISupportInitialize).BeginInit()
        Me.SuspendLayout()
        '
        'Inicio
        '
        Me.Label1.AutoSize = True
        Me.Label1.BackColor = System.Drawing.Color.Transparent
        Me.Label1.ForeColor = System.Drawing.Color.FromArgb(CType(CType(0, Byte), Integer), CType(CType(0, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.Label1.Location = New System.Drawing.Point(12, 368)
        Me.Label1.Name = "Label1"
        Me.Label1.Size = New System.Drawing.Size(48, 15)
        Me.Label1.TabIndex = 1
        Me.Label1.Text = "Inicio..."
        '
        'ProgressBar1
        '
        Me.ProgressBar1.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.ProgressBar1.Location = New System.Drawing.Point(12, 128)
        Me.ProgressBar1.Name = "ProgressBar1"
        Me.ProgressBar1.Size = New System.Drawing.Size(456, 28)
        Me.ProgressBar1.TabIndex = 2
    End Sub
```

Figura 32. Programación de interfaces

1.1.23 ¿Reducir demasiado código?

No hubo necesidad de realizarlo.

1.1.24 ¿Aplicar buenas prácticas de programación?

 Documentación.docx

1.1.25 ¿Existe el modelo del despliegue?

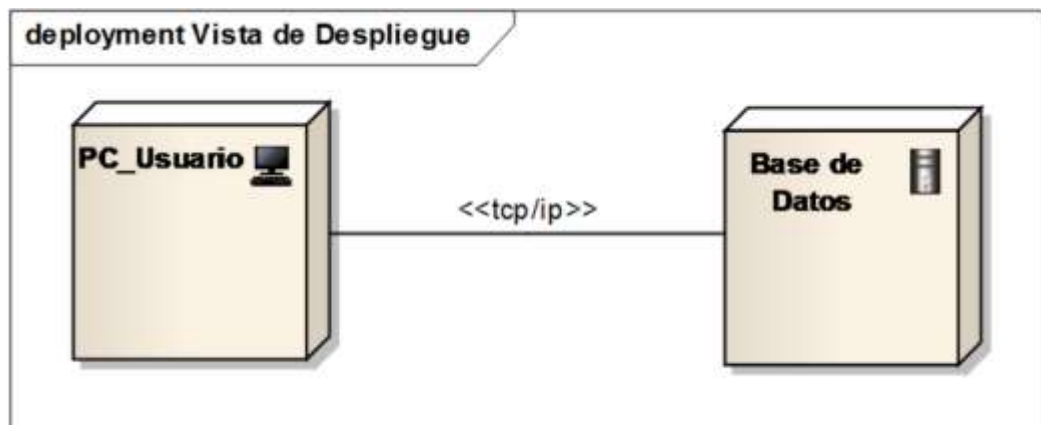


Figura 33. Modelo de despliegue del sistema.

1.1.26 ¿Se utilizó alguna metodología para el proceso de desarrollo del software?

Scrum	
Fase I	Inicio - Análisis
Fase II	Planificación y estimación
Fase III	Implementación
Fase IV	Revisión y retrospectiva
Fase V	Lanzamiento

Figura 34. Proceso de desarrollo scrum.

a.4) Construcción

1.1.27 ¿Se recopilaron correctamente los requisitos para la construcción del software?

No existe el acta de reunión, ya que no hubo necesidad de realizarlo.

1.1.28 ¿Se implementó una arquitectura del Sistema?

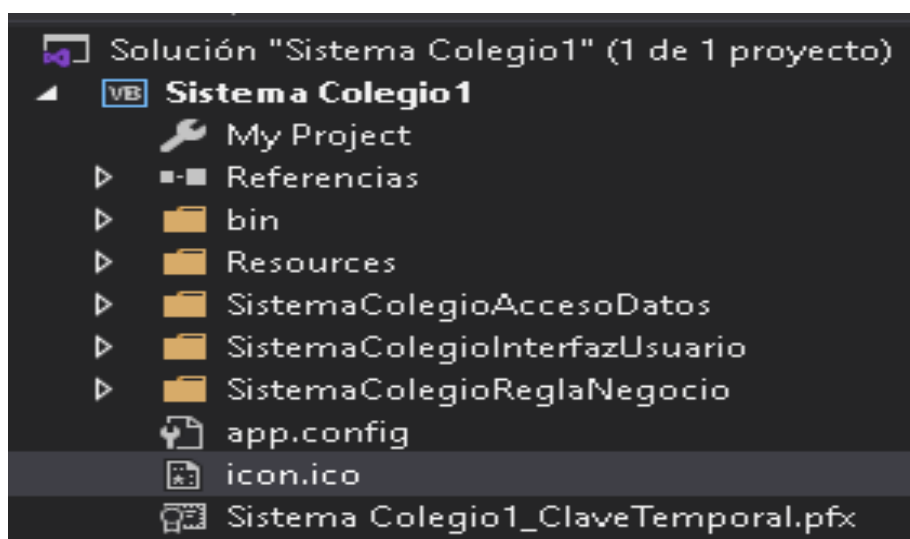


Figura 35. Arquitectura del sistema.

1.1.29 ¿Existe el prototipo del acceso a la aplicación de escritorio?

No existe el prototipo de acceso a la aplicación de escritorio, ya que no hubo necesidad de realizarlo.

1.1.30 ¿Existe el formulario principal de la aplicación de escritorio?



Figura 36. Pantalla principal del sistema.

1.1.31 ¿Existe el panel de control?



Figura 37. Panel de control.

1.1.32 ¿Existe el módulo de alumno?

REGISTRO DE ALUMNOS

Buscar Por: ☐ DNI ☐ Nombre

Dni	Nombre	Ap_Paterno	Ap_Materno	Edad
-----	--------	------------	------------	------

Datos Personales del Alumno

DNI: Nombre:

Apellido Paterno: Apellido Materno:

Edad: Sexo: ☐ Masculino ☐ Femenino Grado De Estudios:

Registrar

Figura 38. Módulo de alumnos.

1.1.33 ¿Existen el módulo de apoderado?

REGISTRO DE APODERADOS

Buscar Por: ☐ DNI ☐ Nombre

Dni	Nombre	Apellido	Edad	Sexo
-----	--------	----------	------	------

Datos Personales del Apoderado

DNI: Nombre:

Apellido Conyugal:

Edad: Sexo: ☐ Esposado ☐ Enviado Grado De Estudios:

Grado De Instrucción:

Registrar

Figura 39. Módulo de apoderado.

1.1.34 ¿Existen el módulo de aulas?

LISTADO DE AULAS

Datos del Aula:

Grado:

Capacidad: Vacantes Libres:

Cantidad de Alumnos:

Figura 40. Modulo aulas.

1.1.35 ¿Existe el módulo de notas?

The screenshot shows a web application window titled "&Buscar" with a red close button. The main header is "REGISTRO DE NOTAS" in a stylized green font. Below the header, there are several input fields and buttons:

- Dni Alumno:** A text input field.
- Seleccione Trimestre:** A dropdown menu with "Primer Bimestre" selected.
- Seleccione Cursos:** A dropdown menu with "Comunicación" selected.
- Descripción:** A dropdown menu with "Seleccione" selected.
- Nota:** A text input field.
- Buttons:** "Agregar" and "Listar" (dark grey buttons), and "Activar" (checkbox).

Below these fields is a table with the following headers: "Curso", "Descripción", and "Nota". The table body is empty.

At the bottom, there are three buttons: "Eliminar", "Registrar", and "Actualizar" (dark grey buttons). To the right of these buttons is a "Promedio" label and a text input field.

Figura 41. Modulo notas.

1.1.36 ¿Existe el módulo de matrículas?

The screenshot shows a web application window titled "FrmMatricula" with a red close button. The main header is "REGISTRO DE MATRÍCULAS" in a stylized green font. Below the header, there are two main sections:

- Registrar Matricula:** A form with the following fields:
 - Codigo:** A text input field.
 - DNI Alumno:** A text input field.
 - Año Escolar:** A text input field.
 - Fecha:** A date picker showing "1/04/2021".
 - Monto:** A text input field.
 - Grado:** A dropdown menu with "Seleccione" selected.
 - Seccion:** A text input field.
 - Vacantes Libres:** A text input field.
 - Partida de Nacimiento:** A checkbox.
 - Certificado de Estudios:** A checkbox.
- Realizar Acciones:** A section with a "Iniciar" checkbox and four buttons: "Registrar", "Actualizar", "Buscar", and "Salir" (dark grey buttons).

Below these sections is a table with the following headers: "Fecha", "Monto", "Grado", "Sección", "Aula", "DNI Alumno", and "Año". The table body is empty.

At the bottom right, there is a "Total Matriculas" label and a text input field showing "0".

Figura 42. Modulo matriculas.

1.1.37 ¿Existe el módulo de docente?

Form9

REGISTRO DE DOCENTE

Datos Personales de Docentes

DNI: Cod Especialidad:

Nombres:

Apellidos Completos:

Edad: Sexo: ☐ Masculino ☐ Femenino Fecha De Nacimiento: 1/04/2021

Dirección: Teléfono:

E-mail:

Realizar Acciones

☐ Iniciar

Buscar Por: ☐ DNI ☐ Nombre

Dni	Nombres	Apellidos	Edad	Sexo	Dirección
25896007	Pedro	Ramirez Ramos	27	M	Av. La Ma
39896238	Doris	Castro Olivera	24	F	Av. Los La
41893237	Walter	Monja Suarez	32	M	Av. Chiclay
43896236	Alex	Gonzales Torres	32	M	La Victoria
45896237	Francisco	Perez Perez	32	M	Urb. Los A

Figura 43. Modulo docente.

1.1.38 ¿Existe el módulo de año escolar?

Form7

REGISTRO DE AÑO ESCOLAR

Año Escolar

Año:

Fecha de Inicio: 1/04/2021

Fecha de Terminó: 1/04/2021

Realizar Acciones

☐ Iniciar

Año	Fecha de Inicio	Fecha de Término

Figura 44. Modulo año escolar.

1.1.39 ¿Se tuvo inconvenientes a último minuto en la implementación de los módulos?

No existe en la documentación los inconvenientes, ya que no hubo necesidad de realizarlo.

a.5) Despliegue

1.1.40 ¿Hay un acta de cierre por módulos implementados?

No existe el acta de cierre, ya que no hubo necesidad de realizarlo.

1.1.41 ¿Se realizó pruebas antes de la entrega del producto?

No existe en el documento pruebas, ya que no hubo necesidad de realizarlo.

1.1.42 ¿Hubo cambios en los requisitos funcionales?

No existe en el documento cambios, ya que no hubo necesidad de realizarlo.

1.1.43 ¿Se documentó los resultados de las pruebas de software?

No existe en el documento resultados de pruebas, ya que no hubo necesidad de realizarlo.

1.1.44 ¿Se realizó el desarrollo del software con controles de seguridad informática?

No existe en el documento la utilización de controles de seguridad informática, ya que no hubo necesidad de realizarlo.

1.1.45 ¿Se implementaron los requisitos funcionales adecuadamente?

No existe en el documento pruebas de la implementación adecuada de los requisitos funcionales, ya que no hubo necesidad de realizarlo.

1.1.46 ¿Se cumplieron con la fecha de los entregables?

No existe en el documento la fecha de los entregables, ya que no hubo necesidad de realizarlo.

1.1.47 ¿Porcentaje de pruebas aprobadas?

No existe en el documento porcentaje de pruebas aprobadas, ya que no hubo necesidad de realizarlo.

1.1.48 ¿Porcentaje de pruebas no aprobadas?

No existe en el documento porcentaje de pruebas no aprobadas, ya que no hubo necesidad de realizarlo.

1.1.49 ¿Porcentaje de pruebas fallados?

No existe en el documento porcentaje de pruebas fallidas, ya que no hubo necesidad de realizarlo.

1.1.50 ¿Si el sistema realiza backup de datos automáticos en un tiempo determinado?

No existe en el documento backup de datos automáticos, ya que no hubo necesidad de realizarlo.

1.1.51 ¿Existe guía de usuario?



GuiaUsuario.docx

1.1.52 ¿El sistema presenta funcionalidad en sus características y aspecto de seguridad?

No existe el documento control de calidad del software, ya que no hubo necesidad de realizarlo.

1.1.53 ¿El sistema presenta facilidad en su uso?

No existe el documento control de calidad del software, ya que no hubo necesidad de realizarlo.

1.1.54 ¿Se capacito a los usuarios?

No existe en el documento la capacitación de los usuarios, ya que no hubo necesidad de realizarlo.

1.1.55 ¿Se configuro correctamente la arquitectura de sistema?

No existe en el documento la prueba de la configuración de la arquitectura, ya que no hubo necesidad de realizarlo.

1.1.56 ¿Los usuarios quedaron satisfechos con el sistema?

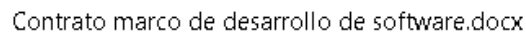
No existe en el documento la prueba de la satisfacción de los usuarios, ya que no hubo necesidad de realizarlo.

1.1.57 ¿Se aplicaron normas ISO de seguridad en el despliegue del sistema?

No existe el documento control de calidad del software, ya que no hubo necesidad de realizarlo.

1.1.58 ¿Se aplicaron normas ISO de seguridad en el despliegue de la base de datos del sistema?

1.1.59 ¿Hay estipulaciones después del cierre del proyecto?



1.1.61 ¿En la integración del software hubo problemas?

1.1.63 ¿Se revisará si las computadoras tienen protección de antivirus?

b) Calidad del proceso de gestión de proyectos

60

SESION DE PROYECTO	ACCIONES	1.1.25	25	Se planifica la gestión de los interesados	Plan de gestión de interesados	x	Max Paez	No hubo necesidad de realizarlo		1	35%
		1.1.26	26	Se realiza la gestión de participación de los interesados	Plan de gestión de interesados	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.27	27	Se adquiere los recursos del proyecto	Plan de adquisiciones	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.28	28	Se priorizan los entregables del plan del proyecto	Plan de gestión de entregables	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.29	29	Se realiza en los aseguramientos de calidad	Plan de gestión de calidad	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.30	30	Se realiza implementación de cambios aprobados	Plan de gestión de cambios	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.31	31	Se realizan acciones aprobadas para situaciones de riesgo	Plan de riesgo	x	Max Paez	No hay documentos probatorios	EVCG-011	5	
		1.1.32	32	Se establecieron los trabajos de equipos con sus funciones	Plan de gestión de equipo	x	Adonai Huaraz	No hubo necesidad de realizarlo	EVCG-012	5	
		1.1.33	33	Se mantiene comunicación y trabajar con los interesados	Plan de gestión de interesados	x	Max Paez	Se encuentra dentro del plan de gestión		1	
		1.1.34	34	Se mantiene un control de las prioridades	Plan de gestión comunicación	x	Adonai Huaraz	Se encuentra dentro del plan de gestión		1	
		1.1.35	35	Se realiza monitoreo y controlando el trabajo del proyecto	Informe de monitoreo y control	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.36	36	Se realiza el control de integrado de cambios	Informe de control integrado	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.37	37	Se realizan los aseguramientos de calidad	Informe de gestión de calidad	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.38	38	Se realiza el control de calidad	Informe de gestión de calidad	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.39	39	Se realiza el control de los alcances	Informe de control de alcances	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.40	40	Se realizan las acciones del programa	Informe de control del programa	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.41	41	Se llevan el control de los costos	Informe de control de costos	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.42	42	Se identifican los riesgos	Plan de gestión de riesgos	x	Adonai Huaraz	Se encuentra dentro del plan de gestión	EVCG-042	5	
		1.1.43	43	Se realizan el análisis cualitativo del riesgo	Plan de gestión de riesgos	x	Max Paez	Se encuentra dentro del plan de gestión	EVCG-043	5	
		1.1.44	44	Se realizan el análisis cuantitativo del riesgo	Plan de gestión de riesgos	x	Adonai Huaraz	Se encuentra dentro del plan de gestión	EVCG-044	5	
MONITORIO + CONTROL		1.1.45	45	Se controlan las adquisiciones	Plan de adquisiciones	x	Max Paez	No hubo necesidad de realizarlo		1	41%
		1.1.46	46	Se realiza el control de las participaciones de los interesados	Informe de control de interesados	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.47	47	Se realiza el control de los riesgos	Plan de riesgo	x	Adonai Huaraz	Se encuentra dentro del plan de gestión	EVCG-047	5	
		1.1.48	48	Se conforma el acta de los entregables evaluados	Doc. Pautaciones Social	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.49	49	Se conforma la propiedad de los entregables a los interesados	Doc. Propiedad	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.50	50	Se conforma el cierre formal y aseguramiento de liberación de responsabilidad formal	Doc. Cierre financiero legal y administrativo	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.51	51	Se conforma el informe final del proyecto	Informe final	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.52	52	Se conforma las acciones aprendidas para evaluar el conocimiento de la organización	Doc. Experiencias	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.53	53	Se conforma los documentos y material del proyecto	Documentos de proyecto	x	Adonai Huaraz	No se realizó	EVCG-053	5	
		1.1.54	54	Se mide la satisfacción del cliente al final del proyecto capturando su retroalimentación	Doc. De satisfacción	x	Max Paez	No hubo necesidad de realizarlo		1	
CIERRE		1.1.55	55	Se conforma el acta de los entregables evaluados	Doc. Pautaciones Social	x	Max Paez	No hubo necesidad de realizarlo		1	11%
		1.1.56	56	Se conforma la propiedad de los entregables a los interesados	Doc. Propiedad	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.57	57	Se conforma el cierre formal y aseguramiento de liberación de responsabilidad formal	Doc. Cierre financiero legal y administrativo	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.58	58	Se conforma el informe final del proyecto	Informe final	x	Adonai Huaraz	No hubo necesidad de realizarlo		1	
		1.1.59	59	Se conforma las acciones aprendidas para evaluar el conocimiento de la organización	Doc. Experiencias	x	Max Paez	No hubo necesidad de realizarlo		1	
		1.1.60	60	Se conforma los documentos y material del proyecto	Documentos de proyecto	x	Adonai Huaraz	No se realizó	EVCG-053	5	
		1.1.61	61	Se mide la satisfacción del cliente al final del proyecto capturando su retroalimentación	Doc. De satisfacción	x	Max Paez	No hubo necesidad de realizarlo		1	

Figura 45. Calidad de proceso de gestión.

b.1) Inicio

2.1.1. Se ha realizado el acta de constitución.

Si se realizó el acta de constitución del proyecto sistema colegio PSC.

Contrato marco de desarrollo de software

REUNIDOS

14 de agosto de 2020

DE UNA PARTE:

Shalom Adonai Huaraz Morales (en adelante, EMPRESA DESARROLLADORA) con 48123639, con domicilio en mz. H lt. 3 Los Olivos, de Lima y en su nombre, actuando en calidad de Lider del proyecto.

Y DE OTRA:

Honores del Zapallal (en adelante, EMPRESA CLIENTE), con domicilio en mz D lt 5 Puente Piedra, de Lima, y en su nombre y representación D. Pedro Estrada, actuando en calidad de director.

Los contratantes se reconocen reciprocamente, en el carácter en que intervienen, plena capacidad

Figura 46. Acta de constitución del proyecto PSC.

2.1.2. Se ha firmado el acta de constitución por el patrocinado.

Si se realizó la firma de constitución del proyecto sistema colegio PSC.

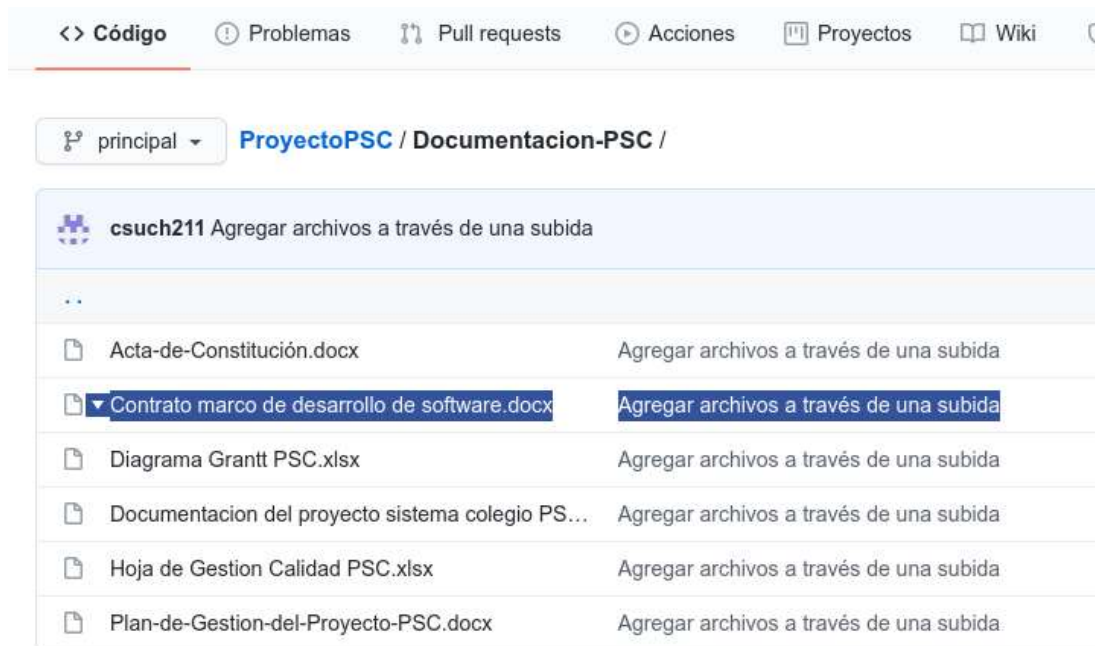


Figura 47. Evidencia documentación del proyecto.

2.1.3. Existe diagrama de caso de uso.



Figura 48. Diagrama de caso de uso.

b.2) Planificación.

2.1.4. Se planifico la gestión del alcance del proyecto.

No se realizó la planificación de alcance del proyecto.

2.1.5. Se recopilamos los requisitos.

Los requisitos del proyecto están definidos en la documentación del proyecto sistema colegio.

Item	Lista de requerimientos inicial
1	El administrador desea agregar, quitar o modificar datos de los usuarios al sistema.
2	El administrador desea hacer consultas fácilmente
3	El administrador desea crear el año escolar
4	El administrador desea que existan niveles de acceso
5	Se desea una interfaz fácil e intuitiva
6	El administrador desea registrar la información académica
7	La secretaria desea registrar la información académica
8	El docente desea registrar calificaciones
9	El administrador desea actualizar la información académica
10	La secretaria desea hacer consultas fácilmente
11	La secretaria desea actualizar la información académica
12	El docente desea hacer consultas fácilmente
13	El docente desea actualizar las calificaciones
14	Se desea una consistente Base de datos

Figura 49. Requerimientos iniciales del sistema.

2.1.6. Se definieron los alcances del proyecto.

El alcance del proyecto está definido en la documentación del proyecto sistema colegio.

5. Alcance

Incluir la tecnología de desarrollo de aplicaciones de escritorio facilitará una mejor eficiencia en el centro escolar ya que podrán tener acceso a los registros académicos de una manera más rápida y eficiente, ya que la implementación de este sistema agilizará algunos procesos que se realizaban de manera manual como los registros de alumnos y matrículas. En el caso de los centros de estudios de primaria el impacto de la tecnología aun no lo han puesto en marcha es por ello que decidimos en tomar este tema para ayudar al sistema educativo a que en las escuelas tomen esta idea y que asistan en forma permanente las operaciones requeridas por los centros de estudio.

5

- El sistema permitirá mejorar la atención al público y por ende agilizar el proceso de inscripción de matrículas.
- La implementación del sistema escolar, agilizará la manera de buscar información detallada de alumnos, matrícula y maestros.
- Brindamos al usuario este sistema como una herramienta para mejorar los diversos procesos del día a día que se realizan en el centro escolar.

Figura 50. Alcance del proyecto.

La figura muestra el EDT del proyecto sistema colegio.

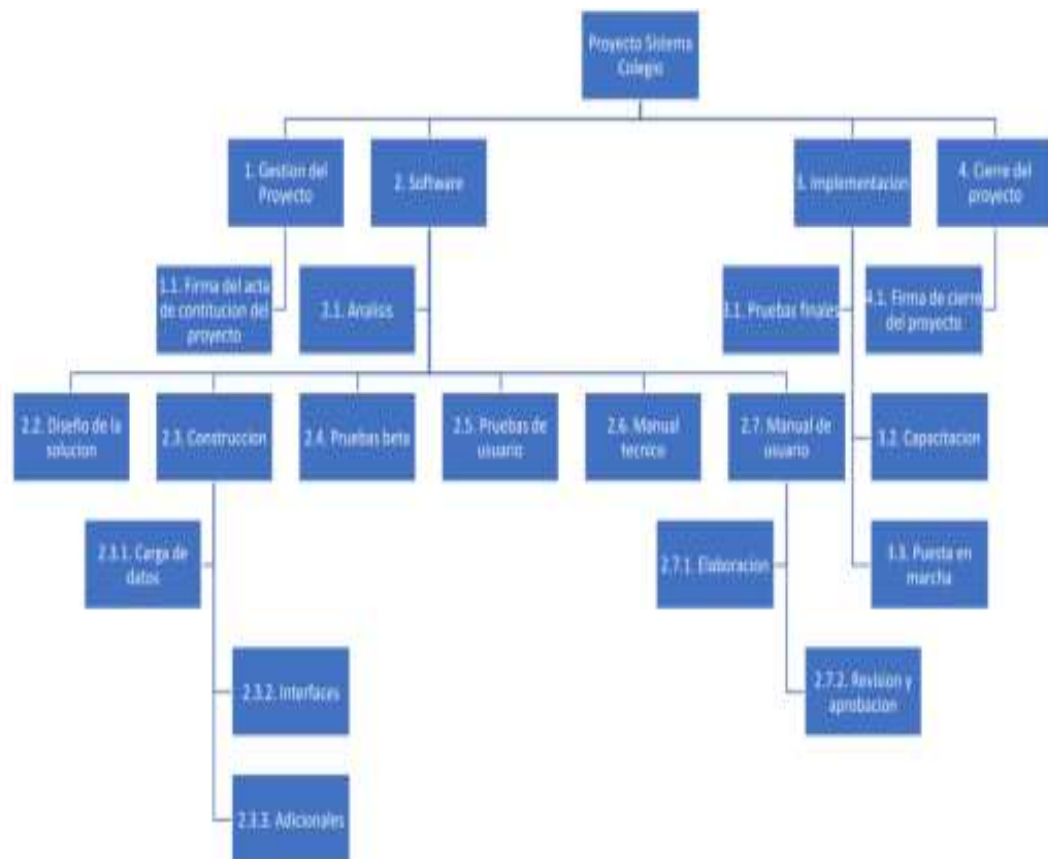


Figura 51. EDT.

2.1.8. Se realizaron la validación de los alcances.

En el documento de gestión del proyecto no se menciona.

2.1.9. Se planifico la gestión del cronograma.

[illegible]

Figura 52. Cronograma del PSC.

2.1.10. Se definieron las actividades.

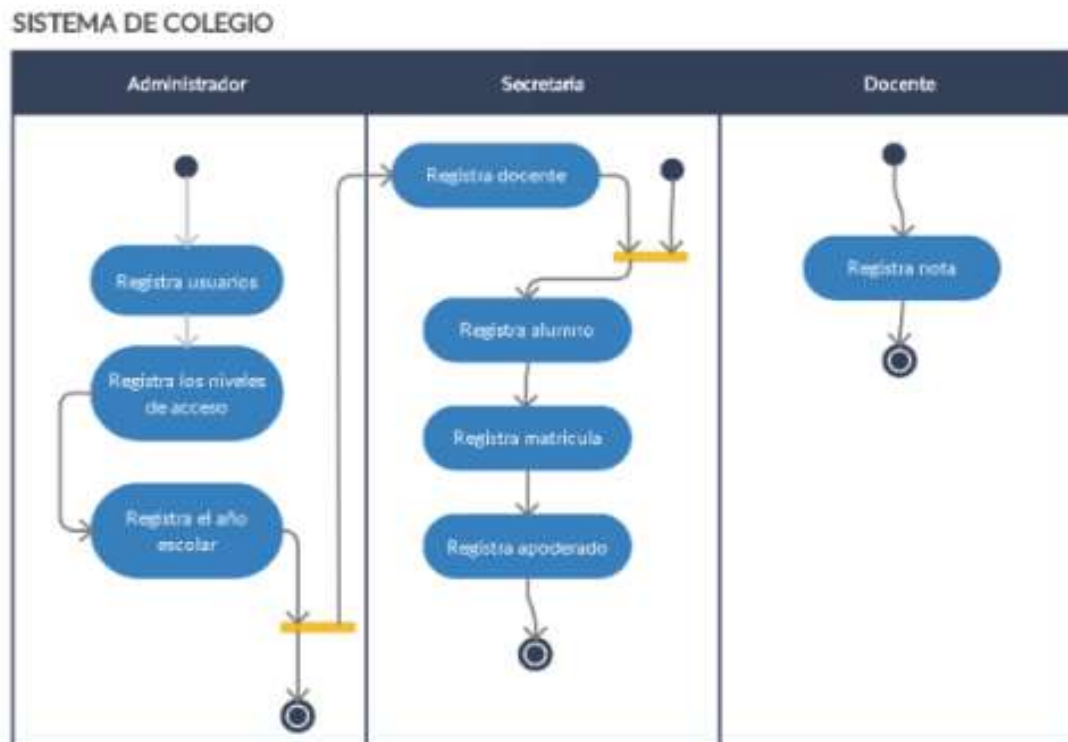


Figura 53. Actividades del negocio.

2.1.11. Se secuenciaron las actividades.



Figura 54. Diagrama de secuencias actualizar matricula.

2.1.12. Se estimaron los recursos de las actividades.

No se realizó la estimación de los recursos de las actividades.

2.1.13. Se estimaron la duración de las actividades.

Nombre de la tarea	Fecha de inicio	Fecha de finalización	Asignado
Proyecto Sistema Colegio	15.08.2020	15.12.2020	Max Paucar
1. Gestion del Proyecto	15.08.2020	15.08.2020	Adonai Huara
1.1. Firma del acta de contit	15.08.2020	15.08.2020	Max Paucar
2. Software	17.08.2020	04.12.2020	Adonai Huara
2.1. Analisis	17.08.2020	20.08.2020	Max Paucar
2.2. Diseño de la solucion	21.08.2020	24.08.2020	Adonai Huara
2.3. Construccion	25.08.2020	19.11.2020	Max Paucar
2.3.1. Carga de datos	25.08.2020	14.10.2020	Adonai Huara
2.3.2. Interfaces	15.10.2020	06.11.2020	Max Paucar
2.3.3. Adicionales	09.11.2020	19.11.2020	Adonai Huara
2.4. Pruebas beta	20.11.2020	24.11.2020	Max Paucar
2.5. Pruebas de usuario	25.11.2020	01.12.2020	Adonai Huara
2.6. Manual tecnico	02.12.2020	02.12.2020	Max Paucar

Figura 55. Duración de las actividades.

2.1.14. Se desarrollo el cronograma.

Nombre de la tarea	Fecha de inicio	Fecha de finalización	Asignado	Estado	15.08.2020	16.08.2020	17.08.2020	18.08.2020	19.08.2020	20.08.2020	21.08.2020	22.08.2020	23.08.2020	24.08.2020	25.08.2020	26.08.2020	27.08.2020
Proyecto Sistema Colegio	15.08.2020	15.12.2020	Max Paucar	Terminado													
1. Gestion del Proyecto	15.08.2020	15.08.2020	Adonai Huara	Terminado													
1.1. Firma del acta de contit	15.08.2020	15.08.2020	Max Paucar	Cerrado													
2. Software	17.08.2020	04.12.2020	Adonai Huara	Terminado													
2.1. Analisis	17.08.2020	20.08.2020	Max Paucar	Terminado													
2.2. Diseño de la solucion	21.08.2020	24.08.2020	Adonai Huara	Terminado													
2.3. Construccion	25.08.2020	19.11.2020	Max Paucar	Terminado													
2.3.1. Carga de datos	25.08.2020	14.10.2020	Adonai Huara	Terminado													
2.3.2. Interfaces	15.10.2020	06.11.2020	Max Paucar	Terminado													
2.3.3. Adicionales	09.11.2020	19.11.2020	Adonai Huara	Terminado													
2.4. Pruebas beta	20.11.2020	24.11.2020	Max Paucar	Terminado													
2.5. Pruebas de usuario	25.11.2020	01.12.2020	Adonai Huara	Terminado													
2.6. Manual tecnico	02.12.2020	02.12.2020	Max Paucar	Terminado													

Figura 56. Cronograma del proyecto.

2.1.15. Se planificaron la gestión de costes.

La documentación no indica nada al respecto.

2.1.16. Se estimaron los costes.

La documentación no indica nada al respecto.

2.1.17. Se determinaron los presupuestos.

La documentación no indica nada al respecto.

2.1.18. Se planificaron la gestión de recursos humanos.

La documentación no indica nada al respecto.

2.1.19. Se planifico la gestión de comunicaciones.

Se encuentra en el plan de gestión del proyecto sistema colegio.

Es importante recordar que el plan de comunicaciones considerará al propio equipo de trabajo como una serie de stakeholders de gran importancia para conseguir el éxito en el proyecto.

Tabla 2. Matriz de Comunicaciones del Proyecto

NOMBRE DEL PROYECTO				SIGLAS DEL PROYECTO				
Sistema Colegio				PSC				
INFORMACIÓN	CONTENIDO	FORMATO	NIVEL DE DETALLE	RESPONSABLE DE COMUNICAR	GRUPO RECEPTOR	METODOLOGÍA O TECNOLOGÍA	FRECUENCIA DE COMUNICACIÓN	CODIGO DE ELEMENTO O WBS
Acta de Proyecto	Contiene a todos los participantes en el proyecto	Papel	Alto	Cliente	Director de proyecto	Escrito	Unica vez	1.1
Alcance preliminar	Contiene hasta donde va abarcar el proyecto inicialmente	Papel	Normal	Director de proyecto	Equipo de proyecto	Escrito	Unica vez	1.1
Reporte del avance	Contiene un informe del avance del proyecto	Papel	Normal	Director de proyecto	Equipo de proyecto y cliente	Escrito	Semanal	1.1
Plan de gestión	Contiene todas las actividades a realizarse para la realización del proyecto	Papel	Alto	Director de proyecto	Equipo de proyecto	Escrito	Unica vez y ante cambio	1.1
Plan de gestión de	Contiene hasta donde va abarcar el proyecto y	Papel	Alto	Director de proyecto	Equipo del proyecto	Escrito	Unica vez y ante cambio	1.1

Figura 57. Plan de gestión de comunicaciones PSC.

2.1.20. Se gestionaron las comunicaciones.

El documento no indica su realización.

2.1.21. Se planificaron la gestión de riesgo.

Se encuentra en el plan de gestión del proyecto sistema colegio.

Riesgos		Categoría	Probabilidad	Impacto	Amortiguación del Impacto
1	La estimación del tamaño del software puede ser significativamente alta.	TP	35	3	Definir alcances posibles de realizar.
2	La estimación del tamaño del software puede ser significativamente alta.	TP	40	3	Tratar de que el número de programas, archivos y transacciones sea bajo.
3	El tamaño de la base de datos creada o empleada por el software puede ser significativamente alto.	TP	45	2	Verificar si se ha realizado un correcto análisis del diseño de las tablas y que cumpla con todas las expectativas del software.
4	Nivel de satisfacción del usuario final.	IN	35	3	Realizar entrevistas y explicarle detalladamente al usuario final que el sistema no perjudicará su labor sino que le traerá beneficios.
5	Cambios significativos en los requerimientos.	RC	40	2	Tener una comunicación directa con los usuarios finales para así disminuir el riesgo de cambios en los requerimientos.
6	Ausencia de participación de los usuarios.	RC	35	3	Tratar de que los usuarios finales participen durante el desarrollo del software.
7	Riesgos del medio ambiente (Ej. Falta de información sobre políticas internas)	PS	10	3	Solicitar a los Departamentos y Secciones toda la información concerniente a las políticas y procedimientos internos del Colegio.
8	Riesgos de interconexión en la red LAN.	TC	10	4	Evaluar y probar la interconectividad entre la arquitectura de red local y las PCs de los usuarios.
9	Falta de conocimiento de las herramientas de programación o uso de nuevas tecnologías.	ED	10	3	Tener manuales sobre las herramientas usadas durante el proyecto.
10	Falta de disponibilidad de herramientas de análisis, diseño y programación.	ED	15	3	Tener un backup de los instaladores de herramientas que se van a usar durante el desarrollo del proyecto.
11	Capacitación deficiente de los usuarios finales.	PP	10	4	Realizar un cronograma adecuado de capacitación del sistema para los usuarios finales.

Figura 58. Plan de Comunicaciones.

2.1.22. Se planificaron la gestión de calidad.

Se encuentra en el plan de gestión del proyecto sistema colegio.

Fase	Entregable	Parámetro de Calidad	Acción Preventiva	Prueba de Inspección
Procedimientos	(2.2) Piloto Operativo	Completo y detallado	(2.2.1) Revisión detallada del Plan de Piloto Operativo Aprobación de Plan de Piloto por usuarios y Comité Ejecutivo	Revisión y aprobación de Informe de Resultados
	(2.3) Manual de Procedimientos	Calidad y veracidad	(2.3.1) Revisión previa de estándares y manuales existentes. Revisión de Informe del Piloto Operativo y construcción de Índice del Manual	Revisión y aprobación de Índice preliminar Revisión cruzada de Manual Contrastar Manual con Informe de Piloto Operativo Aprobación de Usuarios
Software	(3.1) Análisis	Nivel de detalle, claridad y veracidad	(3.1.1) Revisión previa de Especific. Del Prod. Revisión previa de Informes anteriores	Revisión cruzada de Informe Revisión y aprobación Interna de Informe
	(3.2) Diseño	Modelo Normalizado Nivel de detalle en Prototipo Manual	(3.2.1) Revisión previa de especificación del producto. Revisión previa de estándares y Prototipos de aplicaciones anteriores.	RTF Contrastar especificación del producto con prototipo manual Revisión de Prototipo manual con estándares Aprobación de prototipo manual
	(3.3) Construcción	Usabilidad	(3.3.1) Revisión previa de estándares. Revisión de software existente Preparar glosario de términos del usuario.	Revisión modular del software
		Mantenibilidad	(3.3.1) Revisión de software existente Revisión previa de estándares	RTF
		Auditabilidad	(3.3.1) Revisión previa del modelo Revisión previa de software existente	Revisión modular del software
		Fiabilidad	(3.3.1) Revisión previa del modelo Revisión previa de especificación del producto	Pruebas Beta
	(3.5) Manual de Usuario	Claro y veraz	(3.5.1) Revisión previa del software Revisión previa de estándares y manuales existentes. Revisión previa del glosario de términos	Revisión cruzada Aprobación del usuario
	(3.6) Manual Técnico	Claro y veraz	(3.6.1) Revisión previa de estándares y manuales existentes. R	Revisión cruzada Aprobación interna
Implementación	(5.1) Capacitación	Alicance 100% usuarios Practica y orientada al proceso Satisfacción del usuario y Comités del proceso de capacitación	(5.1.1) Revisión de Informes de capacitaciones anteriores Aprobación del programa de capacitación por Comité Ejecutivo del proyecto y aprobación de lista de usuarios	Exámenes aprobados por 100% de usuarios
	(5.3) Paralelo	Cumplimiento 100% del ciclo de la muestra elegida Satisfacción del usuario de los resultados del paralelo	(5.3.1) Reunión previa de información con Stake Holders del Proyecto para formalizar y comprometer a usuarios Selección y aprobación de muestra por Usuarios y Comité Ejecutivo	Revisión y aprobación de usuarios y Comité de Informe de resultados
	(5.4) Puesta en Marcha	Transparencia del proceso Satisfacción del usuario	(5.4.1) Asegurar participación de recursos asignados al proyecto (Equipo y RRHH) Aprobación de usuarios	Revisión y aprobación de informes previo e implementación
	(5.5) Seguimiento C	Cumplimiento del ciclo completo Satisfacción del usuario	(5.5.1) Asegurar la participación total del(os) recurso(s) de soporte asignado(s) por sistemas	Revisión y aprobación de informe de seguimiento
	(5.6) A/C	Cumplimiento del ciclo completo según manual de procedimientos	(5.6.1) Aprobación de Cartilla de control por usuarios y Comité Ejecutivo	Revisión de informes diarios de A&C

Figura 59. Gestión de la calidad.

2.1.23. Se planificaron las respuestas de los riesgos.

Se encuentra en el plan de gestión del proyecto sistema colegio.

	Riesgos	Categoría	Probabilidad	Impacto	Amortiguación del Impacto
1	La estimación del tamaño del software puede ser significativamente alta.	TP	35	3	Definir alcances posibles de realizar.
2	La estimación del tamaño del software puede ser significativamente alta.	TP	40	3	Tratar de que el número de programas, archivos y transacciones sea bajo.
3	El tamaño de la base de datos creada o empleada por el software puede ser significativamente alto.	TP	45	2	Verificar si se ha realizado un correcto análisis del diseño de las tablas y que cumpla con todas las expectativas del software.
4	Nivel de satisfacción del usuario final.	IN	35	3	Realizar entrevistas y explicarle detalladamente al usuario final que el sistema no perjudicará su labor sino que le traerá beneficios.
5	Cambios significativos en los requerimientos.	RC	40	2	Tener una comunicación directa con los usuarios finales para así disminuir el riesgo de cambios en los requerimientos.
6	Ausencia de participación de los usuarios.	RC	35	3	Tratar de que los usuarios finales participen durante el desarrollo del software.
7	Riesgos del medio ambiente (Ej. Falta de información sobre políticas internas).	PS	10	3	Solicitar a los Departamentos y Secciones toda la información concerniente a las políticas y procedimientos internos del Colegio.
8	Riesgos de interconexión en la red LAN.	TC	10	4	Evaluar y probar la interconectividad entre la arquitectura de red local y las PCs de los usuarios.
9	Falta de conocimiento de las herramientas de programación o uso de nuevas tecnologías.	ED	10	3	Tener manuales sobre las herramientas usadas durante el proyecto.
10	Falta de disponibilidad de herramientas de análisis, diseño y programación.	ED	15	3	Tener un backup de los instaladores de herramientas que se van a usar durante el desarrollo del proyecto.
11	Capacitación deficiente de los usuarios finales.	PP	10	4	Realizar un cronograma adecuado de capacitación del sistema para los usuarios finales.
12	Poca recolección de información.	RC	15	3	En las entrevistas, tomar apuntes de todos los detalles requeridos por los usuarios sin omitir nada. Realizar dichas entrevistas periódicamente.
13	El costo del proyecto supere el presupuesto.	IN	45	2	Elaborar bien el presupuesto para la implementación del sistema.
14	El sistema puede tener fallas durante su operación y funcionamiento.	PP	20	3	Realizar las pruebas necesarias antes de que el sistema salga a producción.
15	El sistema puede tener fallas durante su operación y funcionamiento.	TC	25	2	Configurar y realizar pruebas de conexión entre la base de datos y el sistema.
16	No tener un buen control de estándares.	PS	20	3	Se deberá realizar el control y seguimiento a los estándares empleados para la documentación, análisis, diseño y desarrollo del proyecto.
17	No contar con el apoyo de los jefes de Departamentos y Secciones del HCFAP.	RC	20	3	Notificar a los Departamentos y Secciones del Hospital que si no se cuenta con el apoyo de ellos, el proyecto podría fracasar.
18	Falta de licencias de software.	ED	45	2	Coordinar previamente con la Jefatura del Departamento, la adquisición de licencias de software a utilizar para el desarrollo del proyecto.
19	Servidor de aplicaciones defectuoso.	TC	35	2	Escoger un servidor con las características adecuadas para la implantación del sistema y además que se encuentre en perfecto estado.
20	Mala configuración del Servidor Central.	TC	30	3	Contar con una capacitación adecuada para una configuración óptima del servidor.

Figura 60. Plan de respuesta de control de riesgos.

2.1.24. Se planificaron la gestión de adquisiciones del proyecto.

Se encuentra en el plan de gestión del proyecto sistema colegio.

b) Identificar los roles y responsabilidades de los actores que participaran en el proceso de adquisiciones y suministro

RECURSOS	ROLES	RESPONSABILIDADES
Gerente de proyecto	Aprobación	-Conocer el proceso de adquisiciones. Entender el contenido de cada contrato. -Verificar que los contratos abarquen la totalidad del trabajo requerido por el proyecto. -Identificar y mitigar los riesgos, asignando los recursos apropiados para que no afecten al proyecto. -Adicionar al cronograma, el periodo o tiempo necesario para el proceso de adquisiciones. -Coordinar las conferencias de licitación con los proveedores. -Revisión de las propuestas de los proveedores. -Tomar las acciones legales o administrativas en caso de que un proveedor incumpla el contrato o algún riesgo se materialice. -Firma y aprobación de los contratos y adquisiciones del proyecto
Jefe administrativo de calidad	Evaluación y auditoría	-Conocer y entender los contratos celebrados para el proyecto -Velar por el cumplimiento de los proveedores en cuanto a tiempo, alcance y objetivos para los cuales fueron contratados. -Reportar al gerente de proyecto cualquier incidencia que afecte o no esté acorde a los contratos celebrados
Jefe de desarrollo	Auditoría	-Aprobar los perfiles contratados que estarán bajo su mandato e informar al gerente de proyecto para la realización de los contratos laborales.

Figura 61. Plan de gestión de adquisiciones.

2.1.25. Se planifico la gestión de los interesados.

Se encuentra en el plan de gestión del proyecto sistema colegio.

INFORMACION	FORMATO	MEDIO	DESTINATARIO	FRECUENCIA
Apertura, Puesta en Marcha, Cierre del Proyecto	Actas	Escrito	Stakeholders del Proyecto	Inicio, Final y Puesta en Marcha
Actas de Reunión	Actas	Escrito	Stakeholders del Proyecto	Por cada reunión
Actas del Comité de Coordinación	Actas	Escrito	Stakeholders del Proyecto	Por cada reunión
Actas del Comité Ejecutivo	Actas	Escrito	Stakeholders del Proyecto	Por cada reunión
Plan de Proyecto	Documento Plan de Proyecto	Escrito	Stakeholders del Proyecto	Inicio del Proyecto
Resultados de Pruebas Beta, Usuario, Ensamble y Capacitación	Informes	Escrito	Stakeholders del Proyecto	Por cada reunión
A&C	Informes	Escrito	Stakeholders del Proyecto	Al final de cada auditoría
Auditoría Postimplantación	Informes	Escrito	Stakeholders del Proyecto	Al final de cada auditoría

Figura 62. Plan de Gestión de los interesados.

2.1.26. Se realizo la gestión de participación de los interesados.

En la documentación no se encuentra.

b.3) Ejecución.

2.1.27. Se adquirió los recursos del proyecto.

No hay documento probatorio.

2.1.28. Se produjeron los entregables del plan del proyecto.

No hay documento probatorio.

2.1.29. Se realizaron los aseguramientos de calidad.

No hay documentos que pruebe su realización.

2.1.30. Se realizo implementación de cambios aprobados.

No hay documentos que pruebe su realización.

2.1.31. Se realizaron acciones aprobadas para situaciones de riesgo.

	Riesgos	Categoría	Probabilidad	Impacto	Amortiguación del Impacto
1	La estimación del tamaño del software puede ser significativamente alta.	TP	35	3	Definir alcances posibles de realizar.
2	La estimación del tamaño del software puede ser significativamente alta.	TP	40	3	Tratar de que el número de programas, archivos y transacciones sea bajo.
3	El tamaño de la base de datos creada o empleada por el software puede ser significativamente alto.	TP	45	2	Verificar si se ha realizado un correcto análisis del diseño de las tablas y que cumpla con todas las expectativas del software.
4	Nivel de satisfacción del usuario final.	IN	35	3	Realizar entrevistas y explicarle detalladamente al usuario final que el sistema no perjudicará su labor sino que le traerá beneficios.
5	Cambios significativos en los requerimientos.	RC	40	2	Tener una comunicación directa con los usuarios finales para así disminuir el riesgo de cambios en los requerimientos.
6	Ausencia de participación de los usuarios.	RC	35	3	Tratar de que los usuarios finales participen durante el desarrollo del software.
7	Riesgos del medio ambiente (Ej. Falta de información sobre políticas internas).	PS	10	3	Solicitar a los Departamentos y Secciones toda la información concerniente a las políticas y procedimientos internos del Colegio.
8	Riesgos de interconexión en la red LAN.	TC	10	4	Evaluar y probar la interconectividad entre la arquitectura de red local y las PCs de los usuarios.
9	Falta de conocimiento de las herramientas de programación o uso de nuevas tecnologías.	ED	10	3	Tener manuales sobre las herramientas usadas durante el proyecto.
10	Falta de disponibilidad de herramientas de análisis, diseño y programación.	ED	15	3	Tener un backup de los instaladores de herramientas que se van a usar durante el desarrollo del proyecto.
11	Capacitación deficiente de los usuarios finales.	PP	10	4	Realizar un cronograma adecuado de capacitación del sistema para los usuarios finales.
12	Poca recolección de información.	RC	15	3	En las entrevistas, tomar apuntes de todos los detalles requeridos por los usuarios sin omitir nada. Realizar dichas entrevistas periódicamente.
13	El costo del proyecto supere el presupuesto.	IN	45	2	Elaborar bien el presupuesto para la implementación del sistema.
14	El sistema puede tener fallas durante su operación y funcionamiento.	PP	20	3	Realizar las pruebas necesarias antes de que el sistema salga a producción.
15	El sistema puede tener fallas durante su operación y funcionamiento.	TC	25	2	Configurar y realizar pruebas de conexión entre la base de datos y el sistema.
16	No tener un buen control de estándares.	PS	20	3	Se deberá realizar el control y seguimiento a los estándares empleados para la documentación, análisis, diseño y desarrollo del proyecto.
17	No contar con el apoyo de los jefes de Departamentos y Secciones del HCFAP.	RC	20	3	Notificar a los Departamentos y Secciones del Hospital que si no se cuenta con el apoyo de ellos, el proyecto podría fracasar.
18	Falta de licencias de software.	ED	45	2	Coordinar previamente con la Jefatura del Departamento, la adquisición de licencias de software a utilizar para el desarrollo del proyecto.
19	Servidor de aplicaciones defectuoso.	TC	35	2	Escoger un servidor con las características adecuadas para la implantación del sistema y además que se encuentre en perfecto estado.
20	Mala configuración del Servidor Central.	TC	30	3	Contar con una capacitación adecuada para una configuración óptima del servidor.

Figura 63. Plan de acción de control de riesgos.

2.1.32. Se establecieron los trabajos de equipos con sus funciones.

8.1.2. Definición de ROLES del proyecto en SCRUM

Tabla 8. Definición de Roles del Proyecto

Rol	Personas	Descripción de tareas
<i>Product Owner:</i> Dueño de Producto	Jorge Fernández C., Representante de la empresa o Institución.	Administración del negocio desde la perspectiva del negocio
<i>Scrum Master:</i> Jefe del proyecto	Adonai Huaraz Morales	Gestión y asegurar que el proyecto se lleve a cabo.
<i>Scrum Team:</i> Equipo de desarrollo	Adonai Huaraz Morales Max Ivan Paucar Carrasco	Codificación de las funcionalidades Diseño de los GUI. Pruebas de las funcionalidades.

Figura 64. Definición de roles del proyecto.

2.1.33. Se mantuvo comunicación y trabajo con los interesados.

La documentación no indica nada al respecto.

b.4) Monitoreo y control.

2.1.34. Se realizó un control de las comunicaciones.

No hay documentación al respecto.

2.1.35. Se estuvo monitoreando y controlando el trabajo del proyecto.

En la documentación no hay registro.

2.1.36. Se realizó el control de integrado de cambios.

En la documentación no hay registro.

2.1.37. Se realizaron los seguimientos de calidad.

En la documentación no hay registro.

2.1.38. Se realizó el control de calidad.

En la documentación no hay registro.

2.1.39. Se realizó el control de los alcances.

La documentación no indica nada al respecto.

2.1.40. Se realizaron los controles del cronograma.

La documentación no indica nada al respecto.

2.1.41. Se llevaron el control de los costes.

La documentación no indica nada al respecto.

2.1.42. Se identificaron los riesgos.

Riesgos	Categoría	Probabilidad	Impacto	Amortiguación del Impacto
1 La estimación del tamaño del software puede ser significativamente alta.	TP	35	3	Definir alcances posibles de realizar.
2 La estimación del tamaño del software puede ser significativamente alta.	TP	40	3	Tratar de que el número de programas, archivos y transacciones sea bajo.
3 El tamaño de la base de datos creada o empleada por el software puede ser significativamente alto.	TP	45	2	Verificar si se ha realizado un correcto análisis del diseño de las tablas y que cumpla con todas las expectativas del software.
4 Nivel de satisfacción del usuario final.	IN	35	3	Realizar entrevistas y explicarle detalladamente al usuario final que el sistema no perjudicará su labor sino que le traerá beneficios.
5 Cambios significativos en los requerimientos.	RC	40	2	Tener una comunicación directa con los usuarios finales para así disminuir el riesgo de cambios en los requerimientos.
6 Ausencia de participación de los usuarios.	RC	35	3	Tratar de que los usuarios finales participen durante el desarrollo del software.
7 Riesgos del medio ambiente (Ej. Falta de información sobre políticas internas).	PS	10	3	Solicitar a los Departamentos y Secciones toda la información concerniente a las políticas y procedimientos internos del Colegio.
8 Riesgos de interconexión en la red LAN.	TC	10	4	Evaluar y probar la interconectividad entre la arquitectura de red local y las PCs de los usuarios.
9 Falta de conocimiento de las herramientas de programación o uso de nuevas tecnologías.	ED	10	3	Tener manuales sobre las herramientas usadas durante el proyecto.
10 Falta de disponibilidad de herramientas de análisis, diseño y programación.	ED	15	3	Tener un backup de los instaladores de herramientas que se van a usar durante el desarrollo del proyecto.
11 Capacitación deficiente de los usuarios finales.	PP	10	4	Realizar un cronograma adecuado de capacitación del sistema para los usuarios finales.
12 Poca recolección de información.	RC	15	3	En las entrevistas, tomar apuntes de todos los detalles requeridos por los usuarios sin omitir nada. Realizar dichas entrevistas periódicamente.
13 El costo del proyecto supere el presupuesto.	IN	45	2	Elaborar bien el presupuesto para la implementación del sistema.
14 El sistema puede tener fallas durante su operación y funcionamiento.	PP	20	3	Realizar las pruebas necesarias antes de que el sistema salga a producción.
15 El sistema puede tener fallas durante su operación y funcionamiento.	TC	25	2	Configurar y realizar pruebas de conexión entre la base de datos y el sistema.

Figura 65. Identificación de riesgos.

2.1.43. Se realizaron el análisis cualitativo del riesgo.

OBJETIVOS DEL PROYECTO	ESCALAS RELATIVAS O NUMÉRICAS					
	MUY BAJO 0 - 0.10	BAJO 0.11 - 0.25	MODERADO 0.26 - 0.45	MEDIO ALTO 0.46 - 0.65	ALTO 0.66 - 0.85	MUY ALTO 0.86 - 1.00
ALCANCE	Disminución del Alcance apenas permisible	Impacto menor sobre las funciones secundarias	Impacto menor sobre las funciones principales	Algún impacto sobre áreas funcionales clave.	Impacto significativo sobre la funcionalidad d General	El elemento terminado del proyecto es Inservible
TIEMPO	Insignificante incremento del Tiempo	Incremento Tiempo <5%	Incremento Tiempo 6% - 10%	Incremento Tiempo 11% - 15%	Incremento Tiempo 16% - 25%	Incremento Tiempo >25%
COSTO	Insignificante incremento del costo	Incremento del costo <10%	Incremento costo entre 11% - 20%	Incremento costo entre 21% - 30%	Incremento costo entre 31% - 40%	Incremento del costo >40%

Figura 66. Análisis cualitativo de riesgos.

2.1.44. Se realizaron el análisis cuantitativo del riesgo.

ÍTE M	CATEGORÍA DEL RIESGO	ID	DESCRIPCIÓN DEL RIESGO	ANÁLISIS CUALITATIVO			ANÁLISIS CUANTITATIVO	
				PROBABILIDAD	IMPACTO	SEVERIDAD	IMPACTO (\$)	VALOR MONETARIO ESPERADO (VME)
1	TIEMPO (TP)	TP-1	Cumplimiento de los ítems faltantes	0.20	0.80	0.16	\$ -	\$ -
		TP-2	Estimación de las duraciones de forma precisa	0.35	0.90	0.32	\$ 2,800,000	\$ 980,000
		TP-3	Orden y prioridad de las actividades a desarrollar	0.25	0.70	0.18	\$ -	\$ -
		TP-4	Ejecución de las actividades en un tiempo diferente al programado	0.75	0.90	0.68	\$ 28,287,403	\$ 21,215,552
		TP-5	Aparición de actividades que no están contempladas inicialmente	0.85	0.85	0.72	\$ 6,000,000	\$ 5,100,000
		TP-6	Entrega de materiales y equipos por parte de los proveedores	0.15	0.75	0.11	\$ -	\$ -
2	ECONÓMICOS Y FINANCIEROS (EC)	EC-1	Variación de los costos de los insumos	0.35	0.65	0.23	\$ 402,600,000	\$ 140,910,000
		EC-2	Demora en desembolsos de dinero para la ejecución de actividades.	0.22	0.90	0.20	\$ -	\$ -
		EC-3	Sobrecostos por demoras en la ejecución	0.30	0.85	0.26	\$ 28,287,403	\$ 8,486,221
		EC-4	Variación de los salarios de los profesionales	0.45	0.80	0.36	\$ 12,000,000	\$ 5,400,000
		EC-5	Disponibilidad de materiales	0.50	0.65	0.33	\$ 14,602,105	\$ 7,301,053
3	LEGAL (LG)	LG-1	Obtención de las Pólizas mínimas requeridas antes del inicio de la obra	0.20	0.70	0.14	\$ -	\$ -
		LG-2	Conocimiento de las normas técnicas por parte de los profesionales	0.20	0.90	0.18	\$ -	\$ -
		LG-3	Permisos para el inicio del proyecto	0.30	0.75	0.23	\$ 2,800,000	\$ 840,000
		LG-4	Conocimiento del contrato de ejecución	0.15	0.65	0.10	\$ -	\$ -
		LG-5	Modificaciones en las normas vigentes	0.10	0.95	0.10	\$ -	\$ -
4	GESTIÓN ADMINISTRATIVA (GT)	GT-1	Encontrar proveedores de materiales y equipos indicados	0.60	0.70	0.42	\$ 500,000	\$ 300,000
		GT-2	Repetición de las actividades a realizar	0.20	0.50	0.10	\$ -	\$ -
		GT-3	Transporte de las viviendas desde fábrica al municipio	0.15	0.85	0.13	\$ -	\$ -

		GT-4	Asistencia a los comités de obras	0.20	0.60	0.12	\$ -	\$ -
		GT-5	Perdida de información del proyecto	0.20	0.95	0.19	\$ -	\$ -
5	TÉCNICOS Y DE CALIDAD (TC)	TC-1	Utilización de las guías de análisis y ensayos de materiales	0.25	0.80	0.20	\$ -	\$ -
		TC-2	Disponibilidad de mano de obra calificada	0.60	0.75	0.45	\$ 67,889,767	\$ 40,733,860
		TC-3	Equipos y herramientas de calidad para la ejecución del proyecto	0.30	0.50	0.15	\$ -	\$ -
		TC-4	Cumplimiento de las normas técnicas de construcción	0.10	1.00	0.10	\$ -	\$ -
		TC-5	Información técnica sobre construcciones con materiales similares	0.28	0.70	0.20	\$ -	\$ -
		TC-6	Modificaciones inesperadas en cimentación por solicitud del diseñador	0.10	0.90	0.09	\$ -	\$ -
		TC-7	Diseños deficientes y/o incompletos	0.25	0.95	0.24	\$ 5,833,333	\$ 1,458,333
		TC-8	Especificaciones técnicas incompletas	0.35	0.90	0.32	\$ 833,333	\$ 291,667
		TC-9	Información que aporte a lecciones aprendidas	0.20	0.50	0.10	\$ -	\$ -
		TC-10	Falta de conocimiento del montaje del sistema prefabricado WPC por parte del personal operativo y profesional.	0.60	0.85	0.51	\$ 7,071,851	\$ 4,243,110
6	COMUNICACIÓN (CM)	CM-1	Calidad de los materiales de construcción.	0.20	1.00	0.20	\$ -	\$ -
		CM-2	Palabras con alto nivel técnico.	0.40	0.40	0.16	\$ -	\$ -
		CM-3	Profesionales hagan entender procesos constructivos al personal	0.20	0.95	0.19	\$ -	\$ -
		CM-4	Comunicación de Gerencia con: cliente, proveedores, personal.	0.25	0.70	0.18	\$ -	\$ -
7	EXTERNOS (EX)	EX-1	utilización de los métodos de comunicación adecuados	0.20	0.80	0.16	\$ -	\$ -
		EX-2	Sismos producidos durante la construcción	0.10	0.70	0.07	\$ -	\$ -
		EX-3	Fallas en el terreno no establecidos en	0.30	1.00	0.30	\$ 1,500,000	\$ 450,000

Figura 67. Análisis cuantitativo de riesgos.

2.1.45. Se controlaron las adquisiciones.

En la documentación no hay registro.

2.1.46. Se realizó el control de las participaciones de los interesados.

En la documentación no hay registro.

2.1.47. Se realizó el control de los riesgos.

Se encuentra en la documentación de gestión del proyecto PSC.

Riesgos	Categoría	Probabilidad	Impacto	Amortiguación del Impacto
1 La estimación del tamaño del software puede ser significativamente alta.	TP	35	3	Definir alcances posibles de realizar.
2 La estimación del tamaño del software puede ser significativamente alta.	TP	40	3	Tratar de que el número de programas, archivos y transacciones sea bajo.
3 El tamaño de la base de datos errada o empleada por el software puede ser significativamente alto.	TP	45	2	Verificar si se ha realizado un correcto análisis del diseño de las tablas y que cumpla con todas las expectativas del software.
4 Nivel de satisfacción del usuario final.	IN	35	3	Realizar entrevistas y explicarle detalladamente al usuario final que el sistema no perjudicará su labor sino que le traerá beneficios.
5 Cambios significativos en los requerimientos.	RC	40	2	Tener una comunicación directa con los usuarios finales para así disminuir el riesgo de cambios en los requerimientos.
6 Ausencia de participación de los usuarios.	RC	35	3	Tratar de que los usuarios finales participen durante el desarrollo del software.
7 Riesgos del medio ambiente (Ej. Falta de información sobre políticas internas).	PS	10	3	Solicitar a los Departamentos y Secciones toda la información concerniente a las políticas y procedimientos internos del Colegio.
8 Riesgos de interconexión en la red LAN.	TC	10	4	Evaluar y probar la interconectividad entre la arquitectura de red local y las PCs de los usuarios.
9 Falta de conocimiento de las herramientas de programación o uso de nuevas tecnologías.	ED	10	3	Tener manuales sobre las herramientas usadas durante el proyecto.
10 Falta de disponibilidad de herramientas de análisis, diseño y programación.	ED	15	3	Tener un backup de los instaladores de herramientas que se van a usar durante el desarrollo del proyecto.
11 Capacitación deficiente de los usuarios finales.	PP	10	4	Realizar un cronograma adecuado de capacitación del sistema para los usuarios finales.
12 Poca recolección de información.	RC	15	3	En las entrevistas, tomar apuntes de todos los detalles requeridos por los usuarios sin omitir nada. Realizar dichas entrevistas periódicamente.
13 El costo del proyecto supere el presupuesto.	IN	45	2	Elaborar bien el presupuesto para la implementación del sistema.
14 El sistema puede tener fallas durante su operación y funcionamiento.	PP	20	3	Realizar las pruebas necesarias antes de que el sistema salga a producción.
15 El sistema puede tener fallas durante su operación y funcionamiento.	TC	25	2	Configurar y realizar pruebas de conexión entre la base de datos y el sistema.

Figura 68. Realización del control de riesgos.

b.5) Cierre.

2.1.48. Se confirmó el alcance de los entregables validados.

No existe documentación de confirmación de los entregables validados

2.1.49. Se transfirió la propiedad de los entregables a los interesados

No hay documentación al respecto.

2.1.50. Se comunicó el cierre formal y aseguramiento de liberación de responsabilidad futuras.

No hay documentación referida al cierre formal de responsabilidades futuras.

2.1.51. Se distribuyó el informe final del proyecto.

No hay documento probatorio.

2.1.52. Se recopiló las acciones aprendidas para actualizar el conocimiento de la organización.

No hay documentación al respecto.

2.1.53. Se archivó los documentos y material del proyecto.

La documentación del proyecto PSC está en GitHub.



Figura 69. Documentación del proyecto en GitHub.

2.1.54. Se midió la satisfacción del cliente al final del proyecto capturando su retroalimentación.

No hay documentación que pruebe la existencia.

c) Calidad del código fuente

DOMINIO	SUB-DOMINIO	GITHUB ID	PREGUNTA	ARTIFACTO	SI CUMPLE	NO CUMPLE	REVISOR	COMENTARIO	EVIDENCIA	CALOR CALIDAD POR PREGUNTA	NIVEL DE CALIDAD SUBDOMINIO	NIVEL DE CALIDAD DOMINIO		
CODIGO FUENTE	ESTANDARES DE CODIFICACION	1.1.1	1	Lineas de comentario por funcion	Codigo fuente	X	Max Paucar	El comentario aparece en el codigo fuente	EV-001	5	100%	84%		
		1.1.2	2	Lineas de comentario por modulo	Codigo fuente	X	Adonai Huaraz	El comentario aparece en el codigo fuente	EV-002 (Apoderado), EV-003 (Aulas), EV-004 (Notas), EV-005 (Matricular), EV-006 (Docente), EV-007 (Año escolar)	5				
		1.1.3	3	Inicialización de variables	Codigo fuente	X	Max Paucar	La inicialización de variables aparece en el codigo fuente	EV-008	5				
	FALLAS DE DATOS	1.1.4	4	¿Todas las variables del programa se inician antes de usar sus valores?	Codigo fuente	X	Adonai Huaraz	Todas las variables se inician antes de tener valor, como se muestra en el github	EV-012	5	52%			
		1.1.5	5	¿Todas las constantes tienen nombre?	Codigo fuente		X	Max Paucar	No se requirio el uso de constantes	EV-013			1	
		1.1.6	6	¿La cota superior de los arreglos es igual al tamaño del arreglo o valor + 1?	Codigo fuente	X	Adonai Huaraz	Implementado como muestra la imagen en el github	EV-014	5				
		1.1.7	7	Si se usan cadenas de caracteres, ¿Se asigna explícitamente un delimitador?	Codigo fuente		X	Max Paucar	No se requirio el uso de delimitadores				1	
		1.1.8	8	¿Existe alguna posibilidad de desbordamiento de buffer?	Codigo fuente		X	Adonai Huaraz	No existe porque se esta usando el metodo clear() para borrar la memoria asignada	EV-015			1	
		FALLAS DE CONTROL	1.1.9	9	¿Para cada enunciado condicional, ¿La condicion es correcta?	Codigo fuente	X	Max Paucar	Se uso de forma efectiva las sentencias condicionales	EV-016	5		100%	
	1.1.10		10	¿Hay certeza de que termine cada ciclo?	Codigo fuente	X	Adonai Huaraz	Los ciclos del codigo fuente cumplen su funcion	EV-017	5				
	1.1.11		11	¿Los enunciados compuestos estan correctamente colocados entre parentesis?	Codigo fuente	X	Max Paucar	Se ha respetado la estructura de los codigos compuestos	EV-018	5				
	1.1.12		12	En caso de enunciados, ¿Se justifican todos los casos posibles?	Codigo fuente	X	Adonai Huaraz	Si cumple con la especificacion requerida	EV-019	5				
	1.1.13		13	Si despues de cada caso en los enunciados se requiere un parentesis, ¿este se incluyó?	Codigo fuente	X	Max Paucar	Cada enunciado cuenta con su parentesis respectivo	EV-020	5				
	FALLAS ENTRADA / SALIDA		1.1.14	14	¿Se usan todas las variables de entrada?	Codigo fuente	X	Adonai Huaraz	Si se cumple con la especificacion requerida	EV-021	5		100%	
		1.1.15	15	¿A todas las variables de salida se les asigna un valor antes de que se produzcan?	Codigo fuente	X	Max Paucar	Se cumple con la especificacion requerida	EV-022	5				
		1.1.16	16	¿Entradas inesperadas pueden causar corrupcion?	Codigo fuente	X	Adonai Huaraz	Se cumple con la especificacion requerida	EV-026	5				
	FALLAS DE INTERFAZ	1.1.17	17	¿Se usan todas las variables de entrada?	Codigo fuente	X	Max Paucar	Se cumple con la especificacion requerida	EV-024	5	100%			
		1.1.18	18	¿A todas las variables de salida se les asigna un valor antes de que se produzcan?	Codigo fuente	X	Adonai Huaraz	Se cumple con la especificacion requerida	EV-025	5				
		1.1.19	19	¿Entradas inesperadas pueden causar corrupcion?	Codigo fuente	X	Max Paucar	Los ingresos de datos estan validados	EV-023	5				
	FALLAS DE GESTION DE ALMACENAMIENTO	1.1.20	20	¿Todas las llamadas a funcion y metodo tienen el numero correcto de parametros?	Codigo fuente	X	Adonai Huaraz	Se cumple la cantidad de parametros requerida por funcion	EV-028	5	100%			
		1.1.21	21	¿Los tipos de parametros formal y real coinciden?	Codigo fuente	X	Max Paucar	Si hay coincidencia entre los dos parametros	EV-027	5				
		1.1.22	22	¿Los parametros estan en el orden correcto?	Codigo fuente	X	Adonai Huaraz	Si se cumple con la especificacion requerida	EV-029	5				
		1.1.23	23	Si los componentes acceden a memoria compartida, ¿Tienen el mismo modelo de estructura de memoria compartida?	Codigo fuente	X	Max Paucar	Los componentes al hacer la llamada se cargan en un DataGridView	EV-030	5				
	FALLAS DE GESTION DE EXCEPCION	1.1.24	24	Si se modifica una estructura vinculada, ¿Todos los vinculos se reasignan correctamente?	Codigo fuente	X	Adonai Huaraz			5	80%			
		1.1.25	25	Si se usa almacenamiento dinamico, ¿el espacio se asigno correctamente?	Codigo fuente		X	Max Paucar	No se requirio usar almacenamiento dinamico	EV-031			1	
		1.1.26	26	¿el espacio se cancela explícitamente despues de que ya no se requiere?	Codigo fuente	X	Adonai Huaraz	Se utilizo el metodo Trim para cancelar los espacios en blanco	EV-032	5				
		1.1.27	27	¿Se tomaron en cuenta todas las posibles condiciones de error?	Codigo fuente	X	Max Paucar	Se tomaron las excepciones y las validaciones para los posibles errores	EV-033	5				
	PATRON DE ARQUITECTURA	PATRON MVC	1.1.28	28	Clases de software esta codificada para vista clases de software esta codificada para control clases de software esta codificada para modelo?	Codigo fuente	X	Adonai Huaraz	No cumple con las características del patron MVC	EV-309	1		20%	
PATRON ACME		1.1.29	29	Las sentencias SQL se ejecutan solo en procedimientos almacenados	Procedimientos almacenados en BD	X	Max Paucar	Los procedimientos almacenados se encuentran dentro de la base de datos	EV-010	5	100%			
		1.1.30	30	Script de creacion BD	Scripts en BD	X	Adonai Huaraz	La creacion de la BD se encuentra en el Query	EV-011	5				

Figura 70. Hoja de comprobación de calidad de código fuente.

c.1) Estándares de codificación

1.1.1 Líneas de comentario por función

```
'Función LIMPIAR'
```

2 referencias

```
Public Sub LIMPIAR(Form As RegAl)
    Form.TxtDni.Clear()
    Form.TxtNombres.Clear()
    Form.TxtApPaterno.Clear()
    Form.TxtApMaterno.Clear()
    Form.TxtDireccion.Clear()
    Form.TxtEdad.Clear()
    Form.RbnMasculino.Checked = True
    Form.DateTimePicker1.Value = "01/01/1998"
    Form.TxtDni.Focus()
End Sub
```

Figura 71. Líneas de comentarios de código.

1.1.2 Líneas de comentario por módulo

```
'En esta función se registra al año escolar'
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
    Dim cmd As SqlCommand
    If txtAescolar.Text <> "" Then
        Try
            Abrir()
            cmd = New SqlCommand("REGISTRAR_AÑOESCOLAR", conexion)
            cmd.CommandType = 4
            cmd.Parameters.AddWithValue("@AÑO", txtAescolar.Text)
            cmd.Parameters.AddWithValue("@FECHA_INICIO", DateTimePicker1.Value)
            cmd.Parameters.AddWithValue("@FECHA_TERMINO", DateTimePicker2.Value)
            cmd.Parameters.AddWithValue("@MENSAJE", SqlDbType.VarChar, 100).Direction = 2
            cmd.ExecuteNonQuery()
            Dim MENSAJE As String = cmd.Parameters("@MENSAJE").Value.ToString
            MessageBox.Show(MENSAJE, "Aviso", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1)
            DataGridView1.Rows.Clear()
            Listar()
            Limpiar()
        Catch ex As Exception : MsgBox(ex.Message)
        End Try
        Cerrar()
    Else
        MessageBox.Show("Adviértase de haber llenado todos los campos para poder continuar", "Colegio", MessageBoxButtons.OK, MessageBoxIcon.Information)
    End If
End Sub
```

Figura 72. Comentario de código por módulo.

1.1.3 Inicialización de variables

```
Private Sub BtnNuevo_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BtnNuevo.Click
    Me.Width = 1003
    If RadioButton7.Checked = True Then
        TxtDni.Text = TextBox6.Text
        TextBox6.Text = ""
        TxtNombres.Text = ""
        TxtNombres.Focus()
    Else
        TxtNombres.Text = TextBox7.Text
        TextBox7.Text = ""
        TxtDni.Text = ""
        TxtDni.Focus()
    End If
    BtnActualizar.Visible = False
    BtnRegis.Visible = True
    'DESCOMENTAR BtnRegistrar.Visible = True
End Sub
```

Figura 73. Comentario de código por módulo.

c.2) Fallas de datos

1.1.4 ¿Todas las variables del programa se inician antes de usar sus valores?

```
Dim vacio As String = ""
0 referencias
Private Sub Button3_Click(ByVal sender As
    Me.Close()
End Sub

0 referencias
Private Sub CheckBox1_CheckedChanged(ByVa
    If CheckBox1.Checked = True Then
        DateTimePicker1.Enabled = True
        DateTimePicker2.Enabled = True
        txtAescolar.Text = Date.Now.Year
        txtAescolar.Enabled = True
        ' txtAescolar.Focus()
    Else
        DateTimePicker1.Enabled = False
        DateTimePicker2.Enabled = False
        txtAescolar.Text = vacio
```

Figura 74. Inicialización de variables.

1.1.5 ¿Todas las constantes tienen nombre?

No se usó valores constantes.

1.1.6 ¿La cota superior de los arreglos es igual al tamaño del arreglo o Valor = 1?

```
Sub listar()
    Dim dt As New DataTable
    Dim da As SqlDataAdapter
    Try
        abrir()
        da = New SqlDataAdapter("SELECT * FROM AÑO_ESCOLAR ", conexion)
        da.Fill(dt)
        For i = 0 To dt.Rows.Count - 1
            DataGridView1.Rows.Add(dt.Rows(i).Item(0))
            DataGridView1.Rows(i).Cells(0).Value = dt.Rows(i)(0) & ""
            DataGridView1.Rows(i).Cells(1).Value = dt.Rows(i)(1) & ""
            DataGridView1.Rows(i).Cells(2).Value = dt.Rows(i)(2) & ""
        Next
        'DataGridView1.DataSource = dt
    Catch ex As Exception : MsgBox(ex.Message)
    End Try
    Cerrar()
End Sub
```

Figura 75. Cota superior de arreglos.

1.1.7 ¿Si se usan cadenas de caracteres, ¿se asigna explícitamente un delimitador?

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
    Dim cmd As SqlCommand
    Dim delimitador As String = vbCrLf + vbCrLf
    If txtAescolar.Text <> "" Then
        Try
            abrir()
            cmd = New SqlCommand("REGISTRAR_AESCOLAR", conexion)
            cmd.CommandType = 4
            cmd.Parameters.AddWithValue("@NUM_AÑO", txtAescolar.Text)
            cmd.Parameters.AddWithValue("@FECHA_INICIO", DateTimePicker1.Value)
            cmd.Parameters.AddWithValue("@FECHA_TERMINO", DateTimePicker2.Value)
            cmd.Parameters.Add("@MENSAJE", SqlDbType.VarChar, 100).Direction = 2
            cmd.ExecuteNonQuery()
            Dim MENSAJE As String = cmd.Parameters("@MENSAJE").Value.ToString
            MessageBox.Show(MENSAJE, "Aviso", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1)
            DataGridView1.Rows.Clear()
            listar()
            limpiar()
        Catch ex As Exception : MsgBox(ex.Message)
        End Try
        Continuar()
    Else
        MessageBox.Show("Problema al registrar:" + delimitador + "Asegúrese de haber llenado todos los campos para poder continuar",
    End If
End Sub
```

Figura 76. Uso de delimitador de cadenas.

1.1.8 ¿Existe alguna posibilidad de desbordamiento de buffer?

```
If msj = "Datos Incorrectos" Then
    TXTCLAVE.Clear()
    TXTUSUARIO.Focus()
Else
    Dim f As New MenuPrincipal
    Me.Hide()
    f.ShowDialog()
End If
```

Figura 77. Posibilidad de desbordamiento de buffer.

c.3) Fallas de control

1.1.9 Para cada enunciado condicional, ¿la condición es correcta?

```
If msj = "Datos Incorrectos" Then
    TXTCLAVE.Clear()
    TXTUSUARIO.Focus()
Else
    Dim f As New MenuPrincipal
    Me.Hide()
    f.ShowDialog()
End If
```

Figura 78. Uso condicional.

1.1.10 ¿Hay certeza de que termine cada ciclo?

```
If msj = "Datos Incorrectos" Then
    TXTCLAVE.Clear()
    TXTUSUARIO.Focus()
Else
    Dim f As New MenuPrincipal
    Me.Hide()
    f.ShowDialog()
End If
```

Figura 79. Certeza de terminación de un ciclo.

1.1.11 ¿Los enunciados compuestos están correctamente colocados entre paréntesis?

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim cmd As SqlCommand
    If TxtCodAlumno.Text <> "" And TxtCodApoderado.Text <> "" Then
        Try
            abrir()
            cmd = New SqlCommand("REGISTRAR_PARENTESCO", conexion)
            cmd.CommandType = 4
            cmd.Parameters.AddWithValue("@PARENTESCO", CbxParentesco.SelectedItem)
            cmd.Parameters.AddWithValue("@ONI_ALUMNO", TxtCodAlumno.Text)
            cmd.Parameters.AddWithValue("@ONI_APODERADO", TxtCodApoderado.Text)
            cmd.Parameters.Add("@MENSAJE", SqlDbType.VarChar, 100).Direction = 2
            cmd.ExecuteNonQuery()

            Dim MENSAJE As String = cmd.Parameters("@MENSAJE").Value.ToString
            MessageBox.Show(MENSAJE, "AVISO", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1)
            ListarParentesco()
            LIMPIAR()
        Catch ex As Exception : MsgBox(ex.Message)
        End Try
        Cerrar()
    Else
        MessageBox.Show("Asegúrese de haber llenado todos los campos para poder continuar", "Colegio", MessageBoxButtons.OK, Mess
    End If
End Sub
```

Figura 80. Enunciados correctamente colocados en paréntesis.

1.1.12 En caso de enunciados, ¿se justifican todos los casos posibles?

```
If msj = "Datos Incorrectos" Then
    TXTCLAVE.Clear()
    TXTUSUARIO.Focus()
Else
    Dim f As New MenuPrincipal
    Me.Hide()
    f.ShowDialog()
End If
```

Figura 81. Enunciados justificados.

1.1.13 Si después de cada caso en los enunciados se requiere un paréntesis, ¿éste se incluyó?

```
If TxtNota.Text.Trim <> "" Then
    If CInt(TxtNota.Text) >= 0 And CInt(TxtNota.Text) <= 20 Then
        Verficar = VerficarDescripcion(CbxDescripcion.Text)
        If (Verficar = 0) Then
            DataGridView1.Rows.Add()
            i = DataGridView1.Rows.Count
            DataGridView1.Rows(i - 1).Cells(0).Value = CbxCurso.SelectedValue
            DataGridView1.Rows(i - 1).Cells(1).Value = CbxTrimestre.SelectedValue
            DataGridView1.Rows(i - 1).Cells(2).Value = TxtCodigoAlumno.Text
            DataGridView1.Rows(i - 1).Cells(3).Value = CbxCurso.Text
            DataGridView1.Rows(i - 1).Cells(4).Value = CbxDescripcion.SelectedItem
            DataGridView1.Rows(i - 1).Cells(5).Value = CInt(TxtNota.Text)
        Else
            MsgBox("Tipo de Descripcion Existe")
        End If
    Else
        MsgBox("Nota Incorrecta")
    End If
Else
    MsgBox("Asegurese de Llenar Todos Los Campos")
End If
```

Figura 82. Uso de paréntesis en enunciados.

c.4) Fallas de entrada/ salida

1.1.14 ¿Se usan todas las variables de entrada?

```
Private Sub DateTimePicker1_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles DateTimePicker1.ValueChanged
    Dim FECHANAC As Date
    Dim EDAD As Integer

    FECHANAC = DateTimePicker1.Value
    EDAD = Now.Year - FECHANAC.Year

    If FECHANAC <= "01/01/1991" Or FECHANAC >= "01/12/2001" Then
        MsgBox("Edad No Permitida Para Realizar Una Matricula")
        TxtEdad.Clear()
        DateTimePicker1.Value = "01/02/2000"
    Else
        TxtEdad.Text = CStr(EDAD)
    End If
    'DateTimePicker1.Value = Now
End Sub
```

Figura 83. Uso de variables de entrada.

1.1.15 ¿A todas las variables de salida se les asigna un valor antes de que se produzcan?

```
Sub MostrarPromedio()  
    Dim i As Integer  
    Dim Prom As Decimal  
    Dim Suma As Decimal  
    For i = 0 To DataGridView1.Rows.Count - 1  
        Suma = Suma + DataGridView1.Rows(i).Cells(5).Value  
        Prom = Math.Round(CDec(Suma / DataGridView1.Rows.Count), 2)  
    Next  
    LblPromedio.Text = CStr(Prom)  
End Sub
```

Figura 84. Asignación de valores a las variables.

1.1.16 ¿Entradas inesperadas pueden causar corrupción?

```
Private Sub btnentrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnentrar.Click  
    Dim usu As String = TXTUSUARIO.Text  
    Dim cont As String = TXTCLAVE.Text  
    If usu = "" Or cont = "" Then  
        MsgBox("Asegúrese de llenar sus Datos Correctamente")  
    Else  
        Dim cmd As New SqlCommand("InicioSesion", conexion)  
        Try  
            abrir()  
            cmd.CommandType = 4  
            With cmd.Parameters  
                .AddWithValue("@Usu", usu)  
                .AddWithValue("@Contra", cont)  
                .Add("@msj", SqlDbType.VarChar, 60).Direction = 2  
            End With  
            cmd.ExecuteNonQuery()  
        Catch  
        End Catch  
    End If  
End Sub
```

Figura 85. Entradas inesperadas.

c.5) Fallas de interfaz

1.1.17 ¿Se usan todas las variables de entrada?

```
'No lo modifique con el editor de código.  
<System.Diagnostics.DebuggerStepThrough()> _  
0 referencias  
Private Sub InitializeComponent()  
    Me.GroupBox2 = New System.Windows.Forms.GroupBox()  
    Me.TextBox10 = New System.Windows.Forms.TextBox()  
    Me.TextBox9 = New System.Windows.Forms.TextBox()  
    Me.Label17 = New System.Windows.Forms.Label()  
    Me.Label16 = New System.Windows.Forms.Label()  
    Me.RadioButton5 = New System.Windows.Forms.RadioButton()  
    Me.RadioButton4 = New System.Windows.Forms.RadioButton()  
    Me.DataGridView1 = New System.Windows.Forms.DataGridView()  
    Me.BtnActualizar = New System.Windows.Forms.Button()  
    Me.BtnRegistrar = New System.Windows.Forms.Button()  
    Me.GroupBox1 = New System.Windows.Forms.GroupBox()  
    Me.DateTimePicker1 = New System.Windows.Forms.DateTimePicker()  
    Me.CbxEstado = New System.Windows.Forms.ComboBox()  
    Me.Label13 = New System.Windows.Forms.Label()  
    Me.Label12 = New System.Windows.Forms.Label()  
    Me.Label11 = New System.Windows.Forms.Label()  
End Sub
```

Figura 86. Uso de variables de entrada.

1.1.18 ¿A todas las variables de salida se les asigna un valor antes de que se produzcan?

```
'No lo modifique con el editor de código.
<System.Diagnostics.DebuggerStepThrough()> _
0 referencias
Private Sub InitializeComponent()
    Me.GroupBox2 = New System.Windows.Forms.GroupBox()
    Me.TextBox10 = New System.Windows.Forms.TextBox()
    Me.TextBox9 = New System.Windows.Forms.TextBox()
    Me.Label17 = New System.Windows.Forms.Label()
    Me.Label16 = New System.Windows.Forms.Label()
    Me.RadioButton5 = New System.Windows.Forms.RadioButton()
    Me.RadioButton4 = New System.Windows.Forms.RadioButton()
    Me.DataGridView1 = New System.Windows.Forms.DataGridView()
    Me.BtnActualizar = New System.Windows.Forms.Button()
    Me.BtnRegistrar = New System.Windows.Forms.Button()
    Me.GroupBox1 = New System.Windows.Forms.GroupBox()
    Me.DateTimePicker1 = New System.Windows.Forms.DateTimePicker()
    Me.CbxEstado = New System.Windows.Forms.ComboBox()
    Me.Label13 = New System.Windows.Forms.Label()
    Me.Label12 = New System.Windows.Forms.Label()
    Me.Label11 = New System.Windows.Forms.Label()
End Sub
```

Figura 87. Uso de valor a las variables de salida.

1.1.19 ¿Entradas inesperadas pueden causar corrupción?

```
Private Sub TXTCLAVE_KeyPress(ByVal sender As Object, ByVal e As KeyEventArgs)
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
End Sub

0 referencias
Private Sub TXTUSUARIO_KeyPress(ByVal sender As Object, ByVal e As KeyEventArgs)
    If Char.IsLetter(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
End Sub
```

Figura 88. Entradas inesperadas causan corrupción.

c.6) Fallas de gestión de almacenamiento

1.1.20 ¿Todas las llamadas a función y método tienen el número correcto de parámetros?

```
Dim MENSAJE As String = CMD.Parameters("@MENSAJE").Value.ToString
MessageBox.Show(MENSAJE, "AVISO", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButtons.OK)
LISTAR()
LIMPIAR()
Me.Width = 528
```

Figura 89. Llamada de función, métodos y parámetros

1.1.21 ¿Los tipos de parámetro formal y real coinciden?

```
abrir()
CMD = New SqlCommand("ACTUALIZAR_ALUMNO", conexion)
CMD.CommandType = CommandType.StoredProcedure
CMD.Parameters.AddWithValue("@DNI", TxtDni.Text)
CMD.Parameters.AddWithValue("@NOMBRES", TxtNombres.Text)
CMD.Parameters.AddWithValue("@APE_PATERNO", TxtApPaterno.Text)
CMD.Parameters.AddWithValue("@APE_MATERNO", TxtApMaterno.Text)
CMD.Parameters.AddWithValue("@EDAD", TxtEdad.Text)
CMD.Parameters.AddWithValue("@SEXO", If(RbnMasculino.Checked = True, "M", "F"))
CMD.Parameters.AddWithValue("@DIRECCION", TxtDireccion.Text)
CMD.Parameters.AddWithValue("@FECHA_NACIMIENTO", DateTimePicker1.Value)
CMD.Parameters.Add("@MENSAJE", SqlDbType.VarChar, 100).Direction = 2
CMD.ExecuteNonQuery()

Dim MENSAJE As String = CMD.Parameters("@MENSAJE").Value.ToString
MessageBox.Show(MENSAJE, "AVISO", MessageBoxButtons.OK, MessageBoxIcon.Information)
```

Figura 90. Tipos de parámetros formales y reales coinciden.

1.1.22 ¿Los parámetros están en el orden correcto?

```
abrir()
cmd = New SqlCommand("REGISTRAR_PARENTESCO", conexion)
cmd.CommandType = 4
cmd.Parameters.AddWithValue("@PARENTESCO", CbxParentesco.SelectedItem)
cmd.Parameters.AddWithValue("@DNI_ALUMNO", TxtCodAlumno.Text)
cmd.Parameters.AddWithValue("@DNI_APODERADO", TxtCodApoderado.Text)
cmd.Parameters.Add("@MENSAJE", SqlDbType.VarChar, 100).Direction = 2
cmd.ExecuteNonQuery()

Dim MENSAJE As String = cmd.Parameters("@MENSAJE").Value.ToString
MessageBox.Show(MENSAJE, "AVISO", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButtons.OK)
ListarParentesco()
LIMPIAR()
```

Figura 91. Uso de parámetros en el orden correcto.

1.1.27 ¿Se tomaron en cuenta todas las posibles condiciones de error?

```

CMD.Parameters.AddWithValue("@NOMBRES", TxtNombres.Text.Trim)
CMD.Parameters.AddWithValue("@APE_PATERNO", TxtApPaterno.Text.Trim)
CMD.Parameters.AddWithValue("@APE_MATERNO", TxtApMaterno.Text.Trim)
CMD.Parameters.AddWithValue("@EDAD", TxtEdad.Text.Trim)
CMD.Parameters.AddWithValue("@SEXO", If(RbnMasculino.Checked = True, "M", "F"))
CMD.Parameters.AddWithValue("@DIRECCION", TxtDireccion.Text.Trim)
CMD.Parameters.AddWithValue("@FECHA_NACIMIENTO", DateTimePicker1.Value)
CMD.Parameters.AddWithValue("@MENSAJE", SqlDbType.VarChar, 100).Direction = 2
CMD.ExecuteNonQuery()

Dim MENSAJE As String = CMD.Parameters("@MENSAJE").Value.ToString
MessageBox.Show(MENSAJE, "AVISO", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButtons.OK)
LISTAR()
LIMPIAR()
Me.Width = 528
Catch ex As Exception : MessageBox.Show(ex.Message)
Cerrar()
End Try
Else
    MessageBox.Show("Asegúrese de haber llenado todos los campos para poder continuar", "Colegio", MessageBoxButtons.OK)
    BtnActualizar.Visible = False

```

Figura 94. Posibles condiciones de error a tomar en cuenta.

c.8) Patrón MVC

1.1.28 ¿Clases de software está codificada para vista de clases de software, está codificada para control de clases de software, está codificada para el modelo?

En la codificación del software PSC no se usó el patrón MVC.

c.9) Patrón ACME

1.1.29 Las sentencias SQL se ejecutan solo en procedimientos almacenados.

1.1.30 Script de creación de BD

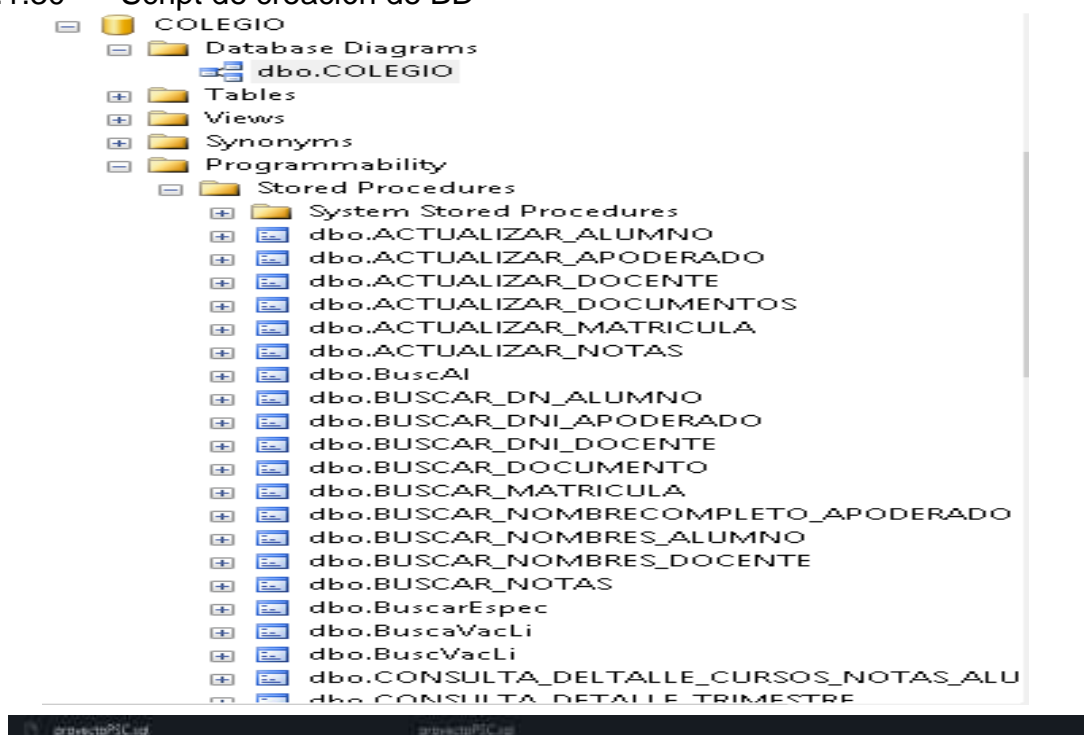


Figura 95. Script de la creación de la base de datos.

3.2.3. Desarrollo de modelos de calidad del producto.

a) Hoja de comprobación de modelos de calidad del producto.

Tabla 4. Modelos de calidad del producto.

Hoja de comprobación del modelo de calidad del proceso										
Característica	Sub característica	N°	Formulación de la pregunta	Evidencia o artefacto	Si cumple (5)	Cumple Parcial (3)	No cumple (0)	Valor de calidad	Recomendación	Sub totales
Adecuación funcional	Compleitud	1	¿Qué tan completa es la implementación de acuerdo a las especificaciones de los requerimientos?	Código fuente	X			5		100%
	Corrección	2	¿Cuán frecuente los usuarios finales encuentran resultados con exactitud inadecuada?	Código fuente	X			5		
	Adecuación	3	¿La aplicación cuenta con las funciones necesarias para que el sistema realice las tareas y objetivos que han sido especificados por el usuario?	Código fuente	X			5		
Eficiencia de desempeño	Comportamiento temporal	4	¿El software cumple con los tiempos de respuesta, procesamiento y ratio de rendimiento que se han establecido?	Código fuente	X			5		100%
	Utilización de recursos	5	¿El software tiene los recursos necesarios para realizar sus funciones bajo las condiciones determinadas?	Código fuente	X			5		
Compatibilidad	Coexistencia	6	¿El software puede compartir recurso con otro software independiente en un mismo entorno?	Código fuente		X		3	Implementar adaptabilidad en la pantalla	60%
	Interoperabilidad	7	¿El software tiene la capacidad de intercambiar información con otros softwares y usarlos en sus procesos?	Código fuente		X		3	Implementar software para intercambiar información con usuarios.	
Usabilidad	Inteligibilidad	8	¿El software tiene la capacidad para permitir al usuario entender si es adecuado para sus necesidades?	Código fuente	X			5		83%
	Aprendizaje	9	¿El software tiene la capacidad de ser entendido por sus usuarios (intuitiva)?	Código fuente	X			5		
	Operabilidad	10	¿El software tiene la capacidad de permitir al usuario operarlo y controlarlo con facilidad?	Código fuente	X			5		
	Protección, errores de usuario	11	¿El software tiene la capacidad de proteger a los usuarios de cometer errores (intuitivo)?	Código fuente	X			5		
	Estética	12	¿Los módulos de interfaz para cada usuario resulta agradable y satisface la interacción con el mismo?	Código fuente	X			5		

	Accesibilidad	13	¿El software tiene la facilidad para ser utilizado por personas con determinadas discapacidades?	Código fuente		X	0	Implementar reconocimiento por voz	
Fiabilidad	Madurez	14	¿Los resultados brindados por el software son realmente fiables?	Código fuente	X		5		50%
	Disponibilidad	15	¿Los datos secundarios (sexo, etc.) que se necesitan para los procesos del software, son mostrados para su selección?	Código fuente	X		5		
	Tolerancia a fallos	16	¿El software presenta la capacidad de operar según la presencia de fallos de hardware o software?	Código fuente		X	0	Implementar equipos alternos	
	Capacidad de recuperación	17	¿El software tiene la capacidad de guardar copias de seguridad de manera automática?	Código fuente		X	0	Implementar backup	
Seguridad	Confidencialidad	18	¿El software tiene la protección necesaria contra el acceso a datos e información no autorizada?	Código fuente	X		5		20%
	Integridad	19	¿El software tiene la capacidad para prevenir accesos o modificaciones no autorizadas a datos o programas del ordenador?	Código fuente		X	0	Implementar seguridad de la información	
	No repudio	20	¿El software permite probar la participación de las diferentes partes de una comunicación teniendo en cuenta el origen y el destino de dichas acciones?	Código fuente		X	0	Implementar comunicación por correo	
	Autenticidad	21	¿El software cuenta con un login de sesión?	Código fuente	X		5		
	Responsabilidad	22	¿El software es responsive en cualquier dispositivo y compatible con cualquier hardware?	Código fuente		X	0	Ser responsive	
Mantenibilidad	Modularidad	23	¿El software está compuesto de componentes discretos que permitan cambios y tengan un impacto mínimo en los demás?	Código fuente	X		5		100%
	Reusabilidad	24	¿El software puede ser utilizado en otro sistema o construcción de otros activos?	Código fuente	X		5		
	Analizabilidad	25	¿El software presenta facilidad para ser evaluado ante determinados cambios y tener diagnóstico de deficiencia o causa de fallos?	Código fuente	X		5		
	Capacidad de ser modificado	26	¿El software permite ser modificado de forma efectiva y eficiente e introducir defectos y degradar el desempeño?	Código fuente	X		5		
	Capacidad de ser probado	27	¿El software permite establecer criterios de pruebas para un sistema o componente?	Código fuente	X		5		

Portabilidad	Adaptabilidad	28	¿El software es responsive en cualquier dispositivo y compatible con cualquier hardware?	Código fuente		X	0	Implementar compatibilidad del SO	67%
	Facilidad de instalación	29	¿El software se puede instalar y desinstalar de manera fácil y rápida sin ningún problema?	Código fuente	X		5		
	Capacidad de ser reemplazado	30	¿El software puede reemplazar de manera óptima otro software cumpliendo el mismo objetivo?	Código fuente	X		5		
Totales		30							76%

3.3. Etapa de ejecución.

3.3.1. Evaluación de calidad del proceso.

Tabla 5. Evaluación de calidad del proceso.

Característica	Sub característica	N°	Hoja de comprobación de evaluación de calidad del proceso				Valor de calidad	Recomendación	Subtotales
			Formulación de la pregunta Evidencia o artefacto	Si cumple (5)	Cumple Parcial (3)	No cumple (0)			
Adecuación funcional	Complejidad	1	¿Qué tan completa es la implementación de acuerdo a las especificaciones de los requerimientos?	X			5		100%
	Corrección	2	¿Cuán frecuente los usuarios finales encuentran resultados con exactitud inadecuada?	X			5		
	Adecuación	3	¿La aplicación cuenta con las funciones necesarias para que el sistema realice las tareas y objetivos que han sido especificados por el usuario?	X			5		
Eficiencia de desempeño	Comportamiento temporal	4	¿El software cumple con los tiempos de respuesta, procesamiento y ratio de rendimiento que se han establecido?	X			5		100%
	Utilización de recursos	5	¿El software tiene los recursos necesarios para realizar sus funciones bajo las condiciones determinadas?	X			5		
Compatibilidad	Coexistencia	6	¿El software puede compartir recurso con otro software independiente en un mismo entorno?		X		3	Implementar adaptabilidad en la pantalla	60%
	Interoperabilidad	7	¿El software tiene la capacidad de intercambiar información con otros softwares y usarlos en sus procesos?		X		3	Implementar software para intercambiar información con usuarios.	
Usabilidad	Inteligibilidad	8	¿El software tiene la capacidad para permitir al usuario entender si es adecuado para sus necesidades?	X			5		83%
	Aprendizaje	9	¿El software tiene la capacidad de ser entendido por sus usuarios (intuitiva)?	X			5		
	Operabilidad	10	¿El software tiene la capacidad de permitir al usuario operarlo y controlarlo con facilidad?	X			5		
	Protección, errores de usuario	11	¿El software tiene la capacidad de proteger a los usuarios de cometer errores (intuitivo)?	X			5		
	Estética	12	¿Los módulos de interfaz para cada usuario resulta agradable y satisface la interacción con el mismo?	X			5		
	Accesibilidad	13	¿El software tiene la facilidad para ser utilizado por personas con determinadas discapacidades?			X	0	Implementar reconocimiento por voz	
Fiabilidad	Madurez	14	¿Los resultados brindados por el software son realmente fiables?	X			5		50%

	Disponibilidad	15	¿Los datos secundarios (sexo, etc.) que se necesitan para los procesos del software, son mostrados para su selección?	X		5		
	Tolerancia a fallos	16	¿El software presenta la capacidad de operar según la presencia de fallos de hardware o software?		X	0	Implementar equipos alternos	
	Capacidad de recuperación	17	¿El software tiene la capacidad de guardar copias de seguridad de manera automática?		X	0	Implementar backup	
Seguridad	Confidencialidad	18	¿El software tiene la protección necesaria contra el acceso a datos e información no autorizada?	X		5		20%
	Integridad	19	¿El software tiene la capacidad para prevenir accesos o modificaciones no autorizadas a datos o programas del ordenador?		X	0	Implementar seguridad de la información	
	No repudio	20	¿El software permite probar la participación de las diferentes partes de una comunicación teniendo en cuenta el origen y el destino de dichas acciones?		X	0	Implementar comunicación por correo	
	Autenticidad	21	¿El software cuenta con un login de sesión?	X		5		
	Responsabilidad	22	¿El software es responsive en cualquier dispositivo y compatible con cualquier hardware?		X	0	Ser responsive	
Mantenibilidad	Modularidad	23	¿El software está compuesto de componentes discretos que permitan cambios y tengan un impacto mínimo en los demás?	X		5		100%
	Reusabilidad	24	¿El software puede ser utilizado en otro sistema o construcción de otros activos?	X		5		
	Analizabilidad	25	¿El software presenta facilidad para ser evaluado ante determinados cambios y tener diagnóstico de deficiencia o causa de fallos?	X		5		
	Capacidad de ser modificado	26	¿El software permite ser modificado de forma efectiva y eficiente e introducir defectos y degradar el desempeño?	X		5		
	Capacidad de ser probado	27	¿El software permite establecer criterios de pruebas para un sistema o componente?	X		5		
Portabilidad	Adaptabilidad	28	¿El software es responsive en cualquier dispositivo y compatible con cualquier hardware?		X	0	Implementar compatibilidad del SO	67%
	Facilidad de instalación	29	¿El software se puede instalar y desinstalar de manera fácil y rápida sin ningún problema?	X		5		
	Capacidad de ser reemplazado	30	¿El software puede reemplazar de manera óptima otro software cumpliendo el mismo objetivo?	X		5		
Totales		30						76%

Nivel de adherencia por característica y general de la evaluación de calidad del proceso.

Tabla 6. Nivel de adherencia en la evaluación de calidad del proceso.

Nivel de adherencia por característica y general				
Característica	Conteo	Total posible	Porcentaje por característica	Porcentaje respecto al general
Adecuación funcional	15	15	100%	10%
Eficiencia de desempeño	10	10	100%	7%
Compatibilidad	6	10	60%	4%
Usabilidad	25	30	83%	17%
Fiabilidad	10	20	50%	7%
Seguridad	10	25	40%	7%
Mantenibilidad	25	25	100%	17%
Portabilidad	10	15	67%	7%
Totales	111	150		76%

Evidencias de la hoja de comprobación en la evaluación de calidad del proceso.

a) Adecuación funcional

EV.001. ¿Qué tan completa es la implementación de acuerdo a las especificaciones de los requerimientos?



Figura 94. Implementación de acuerdo a las especificaciones.

EV.002. ¿Cuán frecuente los usuarios finales encuentran resultados con exactitud inadecuada?



Figura 95. Obtención de exactitud de los resultados

EV.003. ¿La aplicación cuenta con las funciones necesarias para que el sistema realice las tareas y objetivos que han sido especificados por el usuario?



Figura 96. Funciones necesarias del sistema

b) Eficiencia de desempeño

EV.004. ¿El software cumple con los tiempos de respuesta, procesamiento y ratio de rendimiento que se han establecido?



Figura 97. Rendimiento del sistema

EV.005. ¿El software tiene los recursos necesarios para realizar sus funciones bajo las condiciones determinadas?

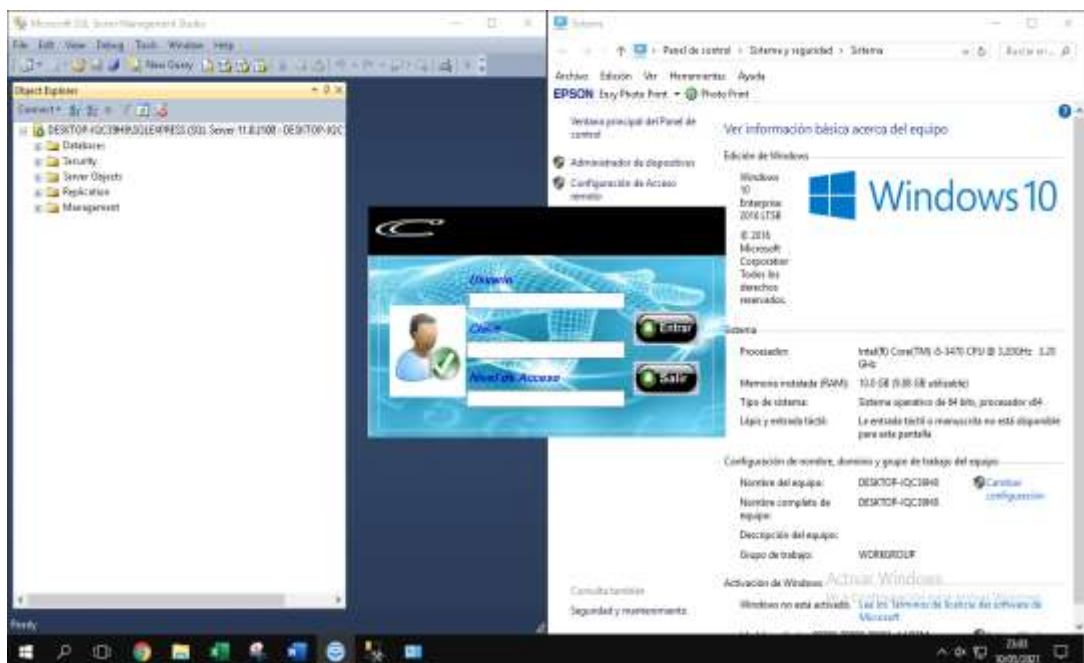


Figura 98. Recursos necesarios para el funcionamiento del sistema.

c) Compatibilidad

EV.006. ¿El software puede compartir recurso con otro software independiente en un mismo entorno?

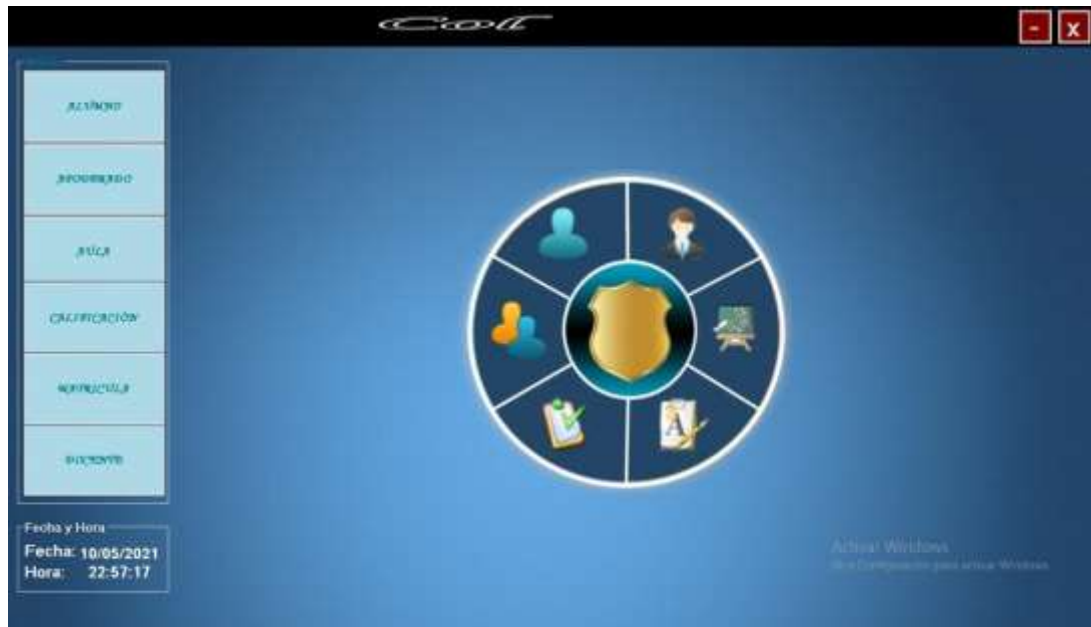


Figura 99. Uso compartido del software.

EV.007. ¿El software tiene la capacidad de intercambiar información con otros softwares y usarlos en sus procesos?



Figura 100. Capacidad del software de compartir la información.

d) Usabilidad

EV.008. ¿El software tiene la capacidad para permitir al usuario entender si es adecuado para sus necesidades?

No, el software no tiene paneles de sugerencia para las necesidades del usuario.

EV.009. ¿El software tiene la capacidad de ser entendido por sus usuarios (intuitiva)?



Figura 101. Capacidad intuitiva del software.

EV.010. ¿El software tiene la capacidad de permitir al usuario operarlo y controlarlo con facilidad?



Figura 102. Uso fácil del software.

EV.011. ¿El software tiene la capacidad de proteger a los usuarios de cometer errores (intuitivo)?



Figura 103. Software intuitivo para corrección de errores.

EV.012. ¿Los módulos de interfaz para cada usuario resulta agradable y satisface la interacción con el mismo?



Figura 104. Interfaz de usuario.

EV.013. ¿El software tiene la facilidad para ser utilizado por personas con determinadas discapacidades?



Figura 105. El Software tiene facilidad de uso para discapacitados.

e) Fiabilidad

EV.014. ¿Los resultados brindados por el software son realmente fiables?

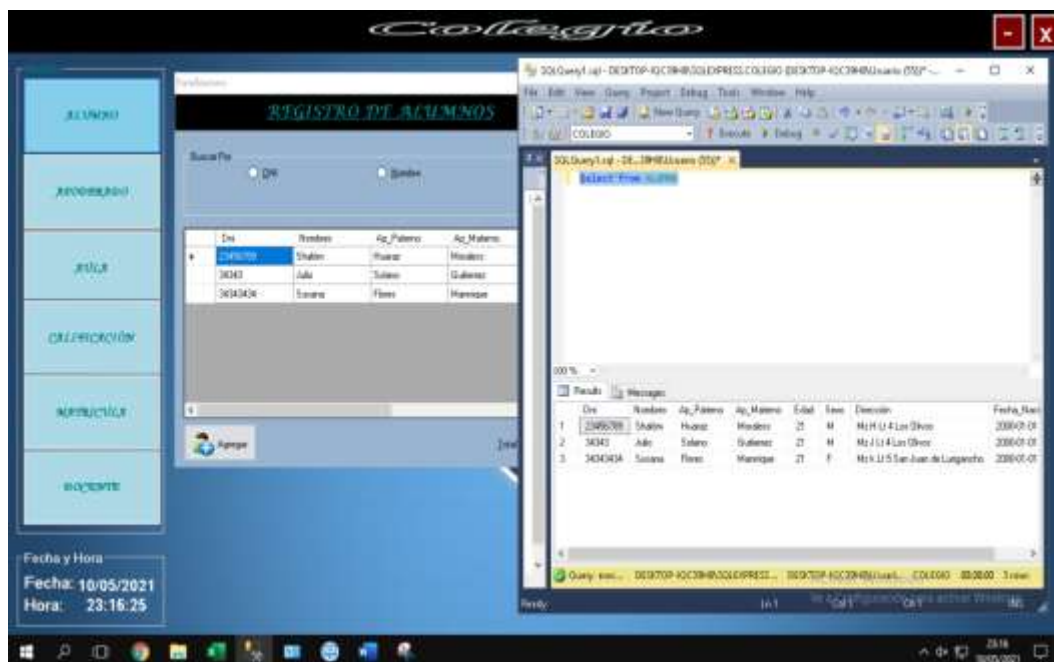


Figura 106. Resultados de software es fiable.

EV.015. ¿Los datos secundarios (sexo, etc.) que se necesitan para los procesos del software, son mostrados para su selección?



Figura 107. Selección de datos secundarios.

EV.016. ¿El software presenta la capacidad de operar según la presencia de fallos de hardware o software?



Figura 108. Software incapaz de funcionar si hay fallos de hardware o software.

EV.017. ¿El software tiene la capacidad de guardar copias de seguridad de manera automática?

No, el software no realiza backup programados.

f) Seguridad

EV.018. ¿El software tiene la protección necesaria contra el acceso a datos e información no autorizada?

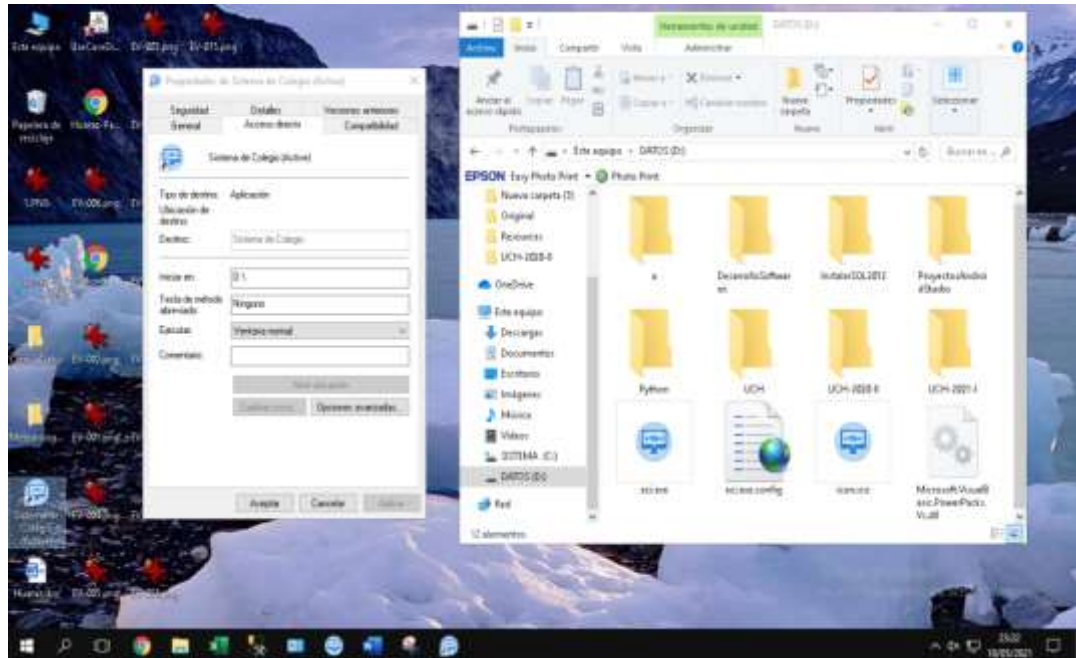


Figura 109. El software no tiene la protección necesaria contra el acceso de datos.

EV.019. ¿El software tiene la capacidad para prevenir accesos o modificaciones no autorizadas a datos o programas del ordenador?



Figura 110. El software tiene la capacidad de prevenir de accesos no autorizados.

EV.020. ¿El software permite probar la participación de las diferentes partes de una comunicación teniendo en cuenta el origen y el destino de dichas acciones?

No, el software no permite la comunicación con otras computadoras.

EV.021. ¿El software cuenta con un login de sesión?



Figura 111. Acceso de usuarios mediante la autenticación de usuario.

EV.022. ¿El software es responsive en cualquier dispositivo y compatible con cualquier hardware?



Figura 112. El software no es portable solo funciona con OS Windows 10.

g) Mantenibilidad

EV.023. ¿El software está compuesto de componentes discretos que permitan cambios y tengan un impacto mínimo en los demás?



Figura 113. El software está compuesto de componentes modulares.

EV.024. ¿El software puede ser utilizado en otro sistema o construcción de otros activos?



Figura 114. El software se puede usar como base de otro software.

EV.025. ¿El software presenta facilidad para ser evaluado ante determinados cambios y tener diagnóstico de deficiencia o causa de fallos?



Figura 115. El software facilita ser evaluado por ser de arquitectura modular.

EV.026. ¿El software permite ser modificado de forma efectiva, eficiente e introducir defectos y degradar el desempeño?



Figura 116. El software permite ser modificado.

EV.027. ¿El software permite establecer criterios de pruebas para un sistema o componente?



Figura 117. El software es factible a las pruebas.

h) Portabilidad

EV.028. ¿El software es responsive en cualquier dispositivo y compatible con cualquier hardware?

No, el software no es responsive ni compatible con cualquier hardware. Es compatible con OS Windows 10 y la arquitectura x64.



Figura 118. el software no es responsive.

EV.029. ¿El software se puede instalar, desinstalar de manera fácil y rápida sin ningún problema?

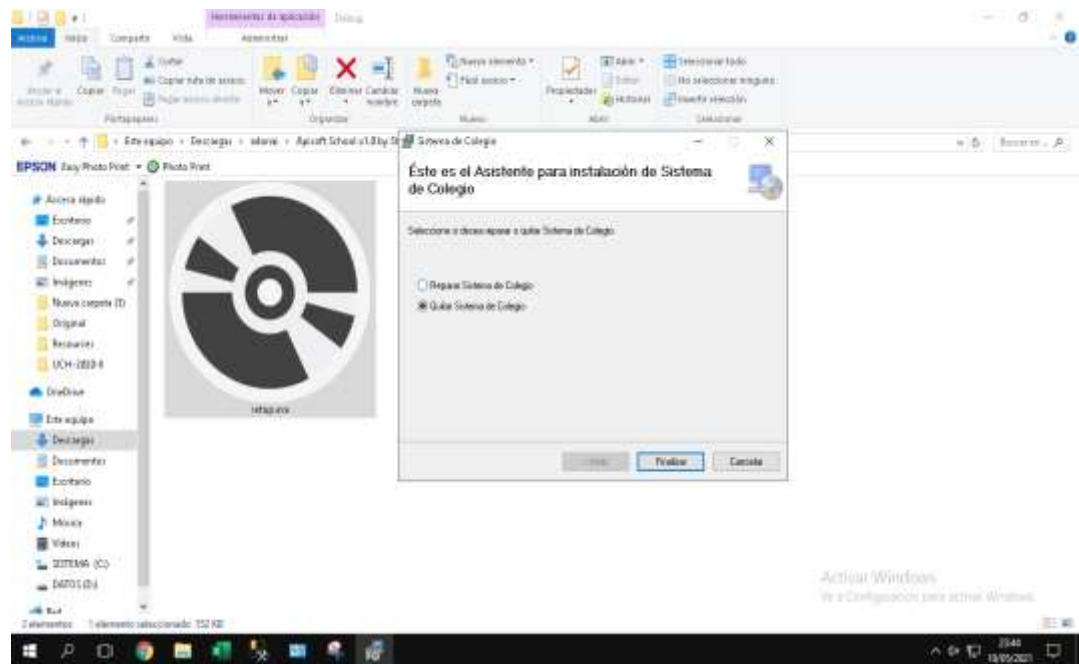


Figura 119. El software es fácil de instalar.

EV.030. ¿El software puede reemplazar de manera óptima otro software cumpliendo el mismo objetivo?



Figura 120. El sistema del sistema colegio muestra el registro de matrículas.

3.3.2. Evaluación de calidad del producto.

Tabla 7. Calidad de uso del producto.

Calidad uso del producto										
Característica	Sub característica	N°	Formulación de la pregunta	Si cumple (5)	Cumple Parcial (3)	No cumple (0)	Valor de calidad	Recomendación	Sub totales	Evidencia
Efectividad	Efectividad	1	¿El software cumple con los requerimientos funcionales establecidos o solicitados?	X			5		20	Ev-001
		2	¿Qué proporción de los objetivos de la tarea es realizado correctamente?	X			5			Ev-002
		3	¿Cuál es la proporción de tareas terminadas?	X			5			Ev-003
		4	¿Cuál es la frecuencia de errores?	X			5			Ev-004
Eficiencia	Eficiencia	5	¿Cuánto tiempo toma el completar una tarea?	X			5		30	Ev-005
		6	¿Qué tan eficientes son los usuarios?	X			5			Ev-006
		7	¿Qué tan efectivo es el usuario en cuanto al costo?	X			5			Ev-007
		8	¿En qué proporción de tiempo desempeña acciones productivas el usuario?	X			5			Ev-008
		9	¿Qué tan productivo es un usuario "sin experiencia" comparándolo con un usuario experto?	X			5			Ev-009
		10	¿El software cumple con las tareas en el tiempo estimado?	X			5			Ev-010
Satisfacción	Utilidad	11	¿Qué tan satisfecho está el usuario con el software?	X			5		18	Ev-011
	Confianza	12	¿Qué tan satisfecho está el usuario con características específicas del software?	X			5			Ev-012
	Placer	13	¿Qué proporción de usuarios potenciales eligen usar el sistema?		X		3	Mejorar el diseño de interfaz de usuario		Ev-013
	Comodidad	14	¿Las funcionalidades del registro de datos están detallado y fáciles de manipular?	X			5			Ev-014
Libertad de riesgo	Mitigación de riesgo económico	15	¿La aplicación a mejorado la eficiencia del proceso y ahorrado recursos?	X			5		30	Ev-015
		16	¿Cuál es la incidencia de daño económico?	X			5			Ev-016
		17	¿Cuál es la incidencia de la corrupción del software?	X			5			Ev-017
	Mitigación de riesgo de salud	18	¿La aplicación ayuda a mantener estable el nivel de estrés del usuario?	X			5			Ev-018
		19	¿Cuál es la frecuencia de problemas de salud de los usuarios que utilizan el producto?	X			5			Ev-019

Mitigación de riesgo medioambiental	20	¿Cuál es la incidencia de riesgo para las personas que utilizan el sistema?	X	5		Ev-020
	21	¿La aplicación contribuye con el ahorro de papel?		0	Implementar un botón de exportar	Ev-021
				X		
Contexto de uso	Integridad del contexto	22	¿El software cumple con los objetivos especificados (seguridad informática las 24 horas los 7 días a la semana) sin ningún riesgo en el contexto de aplicación?	X	5	Ev-022
	Flexibilidad	23	¿El software es flexible a las posibles integraciones o cambios de funcionalidad de un módulo?	X	5	Ev-023
		24	¿El software es flexible a su implementación en una nueva plataforma?	X	5	Ev-024
Totales		24			15	

Nivel de adherencia por característica y general en la evaluación de la calidad en uso.

Tabla 8. Nivel de adherencia en la calidad en uso.

Característica	Conteo	Totales posibles	Porcentaje por característica	Porcentaje respecto al general
Efectividad	20	20	100%	17%
Eficiencia	30	30	100%	25%
Satisfacción	18	20	90%	15%
Libertad de riesgo	30	35	86%	25%
Contexto de uso	15	15	100%	13%
Totales	113	120		95%

Evidencias en la evaluación de calidad del producto por características y subcaracterísticas.

a) Efectividad

a.1) Efectividad

EV-001 ¿El software cumple con los requerimientos funcionales establecidos o solicitados?



Figura 96. Cumplimiento de requerimientos funcionales.

EV-002 ¿Qué proporción de los objetivos de la tarea es realizado correctamente?



Figura 97. Cumplimiento de objetivos.

EV-003 ¿Cuál es la proporción de tareas terminadas?



Figura 98. Cumplimiento de tareas terminadas.

EV-004 ¿Cuál es la frecuencia de errores?



Figura 99. Frecuencia de errores.

b) Eficiencia

b.1) Eficiencia

EV-005 ¿Cuánto tiempo toma el completar una tarea?



Figura 100. Tiempo para completar una tarea.

EV-006 ¿Qué tan eficientes son los usuarios?



Figura 101. Eficiencia de los usuarios.

EV-007 ¿Qué tan efectivo es el usuario en cuanto al costo?



Figura 102. Eficiencia del usuario.

EV-008 ¿En qué proporción de tiempo desempeña acciones productivas el usuario?



Figura 103. Acciones productivas del usuario.

EV-009 ¿Qué tan productivo es un usuario “sin experiencia” comparándolo con un usuario experto?



Figura 104. Productividad del usuario

EV-010 ¿El software cumple con las tareas en el tiempo estimado?



Figura 105. Cumplimientos de tareas.

c) Satisfacción

c.1) Utilidad

EV-011 ¿Qué tan satisfecho está el usuario con el software?

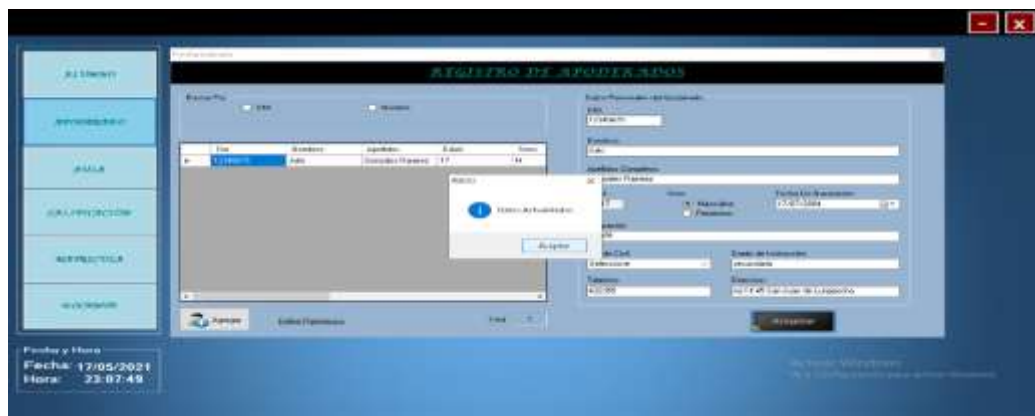


Figura 106. Satisfacción del usuario.

c.2) Confianza

EV-012 ¿Qué tan satisfecho está el usuario con características específicas del software?



Figura 107. Satisfacción del usuario.

c.3) Placer

EV-013 ¿Qué proporción de usuarios potenciales eligen usar el sistema?



c.4) Comodidad

EV-014 ¿Las funcionalidades del registro de datos están detallado y fáciles de manipular?



Figura 108. Funcionalidad intuitiva.

d) Libertad de riesgo

d.1) Mitigación de riesgo económico

EV-015 ¿La aplicación a mejorado la eficiencia del proceso y ahorrado recursos?



Figura 109. Eficiencia del software.

EV-016 ¿Cuál es la incidencia de daño económico?



Figura 110. Software operativo en producción.

EV-017 ¿Cuál es la incidencia de la corrupción del software?



Figura 111. Grado de incidencia de corrupción de software.

d.2) Mitigación de riesgo de salud

EV-018 ¿La aplicación ayuda a mantener estable el nivel de estrés del usuario?



Figura 112. El software es amigable con el usuario.

EV-019 ¿Cuál es la frecuencia de problemas de salud de los usuarios que utilizan el producto?



Figura 113. Software de fácil uso.

EV-020 ¿Cuál es la incidencia de riesgo para las personas que utilizan el sistema?



Figura 114. Riesgos por uso de software.

d.3) Mitigación de riesgo medioambiental

EV-021 ¿La aplicación contribuye con el ahorro de papel?



Figura 115. Automatización de registros.

e) Contexto de uso

e.1) Integridad del contexto

EV-022 ¿El software cumple con los objetivos especificados (seguridad informática las 24 horas los 7 días a la semana) sin ningún riesgo en el contexto de aplicación?



Figura 116. Integridad del software.

e.2) Flexibilidad

EV-023 ¿El software es flexible a las posibles integraciones o cambios de funcionalidad de un módulo?



Figura 117. Software de diseño modular.

EV-024 ¿El software es flexible a su implementación en una nueva plataforma?



Figura 118. Flexibilidad para la implementación.

3.3.3. Desarrollo de pruebas de software.

3.4. Pruebas de caja negra y caja blanca

Etapas de Planificación:

Desarrollo de principios de calidad de software

Desarrollo de modelos de calidad del proceso

Desarrollo de modelos de calidad del producto

Desarrollo de modelos de pruebas de software

Etapas de ejecución:

Evaluación de calidad del proceso

Evaluación de calidad del producto

Desarrollo de pruebas de software

Revisiones de entregables

Etapas de Seguimiento y control:

Seguimiento a la calidad de entregables

Lecciones aprendidas por entregables

Etapas de cierre:

Informe de cierre concluyendo los entregables.

Lecciones aprendidas finales de proyecto

Conclusiones.

10 conclusiones referidas al objetivo, producto, proceso y proyecto. Incluir lecciones aprendidas

10 recomendaciones referidas al proyecto, producto y Proceso

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

En la realización de la calidad de software al Sistema Colegio, iniciamos con los principios de ingeniería del software donde se toman en cuenta el modelo, los requerimientos, el análisis y diseño, la construcción y el despliegue. Como segundo punto se desarrollaron los principios de gestión de proyectos donde se toma en cuenta el ciclo de inicio, planificación, ejecución, monitoreo y control, y cierre. En el tercer punto el principio de código fuente entre ello se toma en cuenta las siguientes fallas: datos, control, entrada y salida, interfaz, gestión de almacenamiento y gestión de ejecución.

En la evaluación de calidad del producto se establecieron preguntas para realizar la comprobación de las características de efectividad, eficiencia, satisfacción, libertad de riesgos y contexto de uso. Ante ello se obtuvo como resultado el 80% de grado de validación en esta etapa del software.

En la medición del proceso del software se establecieron los dominios de ingeniería de software, gestión de proyectos y código fuente que consta cada una de ellas con sus respectivas metas, preguntas, métricas, frecuencias y artefactos. Donde la finalidad es evaluar y tener en consideración las metas que se establecieron para su respectiva verificación y comprobación.

La importancia de la calidad del software, desde la definición del requerimiento hasta su mantenimiento debe realizarse con una calidad total de cada una de las etapas, dependerá de la entrega de un producto que satisfaga las necesidades y exigencias del cliente.

4.2. Recomendaciones

En la calidad de software es importante la documentación para todo el ciclo del desarrollo del software ya que esto realiza su verificación por medio de las hojas de comprobación de gestión, producto y código fuente.

Agregar a la metodología ágil Scrum los diagramas del UML de la metodología del Proceso Racional Unificado (RUP) permite una claridad en el planteamiento de la lógica de la programación del sistema.

Las preguntas establecidas en los diferentes principios de la calidad de software deben ser más precisos y orientados a fortalecer la evaluación del plan.

Las preguntas para la comprobación de las diversas características deberían enfatizar la importancia de los artefactos para que el porcentaje de apego sea más objetivo.

Se propone elaborar planes de contingencia y seguridad para salvaguardar la data de la aplicación.

Es importante que los desarrolladores de software se certifiquen bajo alguna norma o estándar de calidad de software para garantizar un mayor nivel de satisfacción.

REFERENCIAS BIBLIOGRÁFICAS

- Almeida, L. (2015).** *Habilidades interpersonales*. New Jersey: editorial Addison Wesley.
- Barato, J. (2015).** *El director de proyectos a examen: guía de estudio en español para la capacitación del Director de Proyectos*. catalán: Ediciones Díaz de Santos.
- Garzas, Javier. (2012).** *No es lo mismo calidad del producto software, que calidad del proceso software, que calidad de equipo*. (Bajado el 27 de abril del 2019). Recuperado <https://www.javiergarzas.com/2012/08/calidad-del-productosoftware-proceso-equipo.html>
- ISO/IEC 25000 (2005).** *Systems and software engineering – Systems and software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE*, International Organization for Standardization, Geneva, Switzerland
- ISO/IEC 25010 (2005).** *Systems and software engineering – Systems and software product Quality Requirements and Evaluation (SQuaRE) – System and software quality models*, International Organization for Standardization, Geneva, Switzerland
- Maida, Esteban & Pacienza, Julián. (2015).** *Metodologías de desarrollo de software*. Tesis Final de Licenciatura en Sistemas y Computación. UCA. Argentina
- Terra, Gustavo. (2017).** *Pruebas de caja negra y un enfoque práctico*. (Bajado el 24 de junio de 2019). Recuperado <https://testingbaire.com/2017/02/26/pruebas-caja-negra-enfoquepractico/>
- Ibon, Landa. (2015).** *Pruebas unitarias: Cobertura de código*. (Bajado el 24 de junio de 2019). Recuperado <https://geeks.ms/ilanda/2009/03/09/pruebas-unitarias-coberturade-codigo/>

ÍNDICE DE ABREVIATURAS Y SIGLAS

GQM: Goal Question Metric (Objetivo Pregunta Métrica)

ISO: (International Organization for Standardization)

MVC: Model View Controller (Modelo Vista Controlador)

OS: Operating Systems (Sistema Operativo)

PMBOK: Project Management Body of Knowledge

PMI: Project Management Institute

QA: Quality Assurance (Aseguramiento de la calidad)

RUP: Rational Unified Process (Proceso Unificado de Rational)

SQL: Structured Query Language (Lenguaje de consulta estructurada)

GLOSARIO DE TÉRMINOS

A

Acción Correctiva: Acción que se ocupa de eliminar la causa de una no conformidad percibida u alguna situación indeseable.

Acción Preventiva: Acción que se ocupa de eliminar la causa de una no conformidad potencial u alguna situación potencialmente indeseable.

Actas y Certificados: Permitirá administrar las solicitudes de aceptación y fijará a los responsables para asistir y dar respuesta a las solicitudes.

Automatiza: Es atribuir procedimientos automáticos a un objeto, proceso o sistema. Donde se trasladan labores de producción, efectuados usualmente por operadores humanos a una agrupación de elementos tecnológicos.

C

Calidad: Nivel de cumplimiento de las especificaciones del producto. Es el nivel en el cual un sistema o proceso cumple con los requisitos especificados.

Confiabilidad: Es la medición del nivel en que se tiene la facilidad de que la realización de un programa cumpla con su función y con la exactitud solicitada.

Consultas y Búsquedas: Permitirá conseguir la información de la situación presente del consultado.

D

Documento: Recurso de almacenamiento de información. Es un recurso donde se registra la información, por lo común de forma permanente.

E

Eficiencia: Medición del número de recursos de computadora y de código requeridos por un software para que cumpla con sus objetivos.

F

Facilidad de Uso: Es la acción de emplear una gran fuerza física o moral requerida para adquirir el conocimiento de operar un programa y descifrar la información de entrada y de salida.

Flujo: Es el movimiento de algo.

G

Gestión: Hace referencia al acto y a la consecuencia de administrar o gestionar algo. En relación con lo que se trata hay que decir que gestionar es ejecutar trámites que hacen factible la producción de una operación comercial. Administrar,

al contrario de gestionar, contiene la noción de gobernar, disponer guiar y organizar un determinado hecho.

Gestión de Riesgos Trazabilidad: Acto que consiente establecer una relación de la documentación con los sucesos producidos con respecto al riesgo.

I

Implementación: Es la producción de una especificación técnica o algoritmos como un programa, componente software, u algún sistema de cómputo. Varias implementaciones son dadas según a una especificación o un estándar.

Integridad: Es el nivel en que se tiene la facilidad de vigilar el acceso a un software o a los datos por personas no autorizadas

Iteración: Es la acción de volver a realizar un procedimiento con el objetivo de llegar a una meta.

L

Lenguaje de Programación: Es un lenguaje formal diseñado para manifestar procedimientos que tienen la facilidad de ser ejecutados por máquinas como las computadoras. Tienen la facilidad de utilizarse para producir programas que dirigen la manera del comportamiento físico y lógico de una máquina, para manifestar algoritmos con exactitud, o como una forma de comunicación humana.

M

Métricas: Las métricas son mediciones cuantitativas del nivel que un sistema, un elemento del sistema o un procedimiento tiene. Una métrica es utilizada para explicar cómo es un atributo. Asimismo, tienen la facilidad de explicarse como una manera clara como una serie de elementos para la medición. Con lo cual, las métricas son utilizadas como la medición de la calidad para un proyecto.

Modelado del Sistema: Es la agrupación completa de los casos de uso, simbolizan de la funcionalidad propuesta.

P

Proceso: Agrupación de actividades interrelacionados con un propósito específico. Es una serie de etapas que se suceden unos a otros y guardan relación entre sí ejecutando un propósito dado: Por ejemplo, el proceso de desarrollo de software

Proceso de Depuración La depuración no es una prueba, pero se produce como resultado de una prueba. Los resultados se evalúan y se halla la falta de concordancia entre el rendimiento esperado y el real. Por lo común, el proceso de

depuración dará como resultado que la causa. En el último caso, la persona que hace la depuración tiene la facilidad de intuir una causa, crear un caso de prueba para apoyarse en la validez de dicho supuesto y trabajar para la corrección del error.

Pruebas Beta: La prueba beta son las pruebas de software que se realizan en un entorno real. El cliente registra todos los problemas (reales o imaginarios) que se encuentran durante la prueba beta y los reporta al desarrollador periódicamente.

Pruebas de aceptación: Las pruebas de aceptación comparan el comportamiento del sistema con los requisitos del cliente, a fin de determinar si cumple con los requerimientos especificados de la organización.

Pruebas de facilidad de uso: Este proceso evalúa la usabilidad por parte del usuario del software, incluyendo la documentación del usuario.

Pruebas de instalación: Verificar y validar que el sistema se instala apropiadamente en cada cliente. Las pruebas de instalación se pueden ver como pruebas del sistema realizadas en relación con los requisitos de la configuración de hardware.

Pruebas de Integración: Las pruebas de integración es validar la integración de diferentes módulos de una aplicación con el propósito de identificar errores y funcionalidades relacionados con ellos.

Pruebas de Unidad: Son pruebas para comprobar que un fragmento de código funciona correctamente. Las pruebas de unidad se realizan con acceso al código fuente y con el soporte de herramientas de depuración.

Pruebas del Sistema: Las pruebas de sistema se realizan para validar el sistema con los requisitos no funcionales del sistema, como seguridad, rendimiento, exactitud, velocidad y confiabilidad; integrando adecuadamente todos los elementos del sistema y su funcionamiento apropiado.

Pruebas del Software: Es el proceso sistemático de prevención, detección y corrección de defectos de un sistema, según criterios establecidos para asegurarnos la calidad del producto software en relación del comportamiento esperado.

R

Requisito: Los requisitos software son la descripción de las características y funcionalidades de un sistema o componente del sistema para satisfacer un contrato, especificación.

Revisión del Producto: son un conjunto de actividades realizadas como resultado del análisis, diseño y codificación, para detectar y solucionar problemas en un software.

Riesgo: Es la probabilidad de que un evento incierto que, si se produce, tenga un efecto negativo en uno o más objetivos del proyecto.

RUP (Proceso Racional Unificado): Es una metodología de desarrollo de software y junto a UML, constituye la metodología utilizada para el análisis, implementación y documentación de sistemas para una organización.

S

Satisfacción del Cliente: Es una medida de cumplimiento de un producto sobre las expectativas planteadas en que se han cumplido sus requisitos.

SQL: Es un lenguaje de consulta estructurada para el acceso a bases de datos relacionales que permite realizar operaciones de gestión de datos.

T

Tecnología Cliente-Servidor: Es un modelo de diseño de software donde las tareas se reparten entre el cliente que realiza peticiones a un programa y otro llamado servidor que provee los servicios a la solicitud.

V

Validación: El objetivo de la validación es asegurar que el sistema software satisface las expectativas del cliente, para demostrar que el software hace lo que el cliente espera que haga.

Variable: Una variable es un símbolo que permite guardar valores que significan tipos de datos soportados por un lenguaje de programación particular.

Verificación: Es la comprobación del funcionamiento de un software de acuerdo con su especificación, que satisface sus requerimientos funcionales y no funcionales.

Visual Studio: Es un entorno de desarrollo para la generación de aplicaciones de escritorio, web y aplicaciones móviles.

APÉNDICES

