



Facultad 2

Sistema de gestión de las actividades educativas de los estudiantes para su desarrollo integral

**Trabajo de diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

Autores: Jeans Carlos Guzman Dorta
Joan Carlos Galván Díaz

Tutor: Prof. Aux. Madelín Haro Pérez. Ing.

Cotutor: Prof. Asist. Víctor Alejandro Roque Domínguez. Ing.

La Habana, 28 de enero de 2024

“Año 65 de la Revolución”

DECLARACIÓN DE AUTORÍA

Los autores del trabajo de diploma con título “Sistema de gestión de las actividades educativas de los estudiantes para su desarrollo integral”, conceden a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declaran como únicos autores de su contenido. Para que así conste firman la presente, a los 28 de enero de 2024.

Jeans Carlos Guzman Dorta

Firma del Autor

Joan Carlos Galván Díaz

Firma del Autor

Prof. Aux. Ing. Madelín Haro Pérez

Firma del Tutor

Prof. Asist. Víctor Alejandro Roque Domínguez.
Ing.

Firma del Cotutor

Datos del contacto

Autor: Jeans Carlos Guzman Dorta

Correo electrónico: jeanscgd@estudiantes.uci.cu

Autor: Joan Carlos Galván Díaz

Correo electrónico: joancgd@estudiantes.uci.cu

Tutor: Prof. Aux. Madelín Haro Pérez. Ing.

Institución: Universidad de las Ciencias Informáticas

Título: Ingeniera Informática. CUJAE, 2002

Correo electrónico: mharo@uci.cu. Teléfono: 7835 8129

Experiencia como analista, Jefe de Proyectos y Jefe de Capacitación en proyectos productivos. Ha trabajado adiestramiento en Posgrado, ha sido Jefe de Departamento de Disciplina, Profesora Guía, Jefe de Disciplina y de asignatura por más de 13 años y por más de 10, Profesora Principal de Año, de 1ro, 4to y 5to de Ingeniería en Ciencias Informáticas y actualmente de 2do de Ingeniería en Ciberseguridad. Posee la categoría docente Profesor Auxiliar. Tiene preparación posgraduada en temas técnicos y docentes. Sus líneas de trabajo son la gestión de proyectos, la educación universitaria y, actualmente, los proyectos de desarrollo local.

Cotutor: Victor Alejandro Roque Dominguez. Ing.

Institución: Universidad de las Ciencias Informáticas

Título: Ingeniero en Ciencias Informáticas. UCI, 2016

Correo electrónico: varoque@uci.cu. Teléfono: 7835 8133

Desempeño desde graduado en la UJC de la facultad. Profesor de la disciplina de Sistemas Digitales para la carrera de ICI y el programa de ARSI. Profesor principal desde el 2022 de la carrera de Ingeniería en Ciberseguridad. Tribunal de tesis, tutor y oponente para estudiantes de ICI y ARSI. Sus líneas de trabajo son las redes de computadoras.

Agradecimientos

Primeramente, agradezco a mis padres por apoyarme siempre en todas mis decisiones, por confiar en mí, darme aliento y ser mi soporte.

A mis abuelos por ese amor incondicional que siempre me han dado.

Agradecer a mi familia en general por dedicarme su tiempo y brindarme su ayuda de diferentes maneras.

A mi mejor amiga que me ha dado fuerza y no dejó que me rindiera en momentos que pensaba que no podía continuar.

A mis amigos del preuniversitario que a pesar de los años se han mantenido a mi lado y se han preocupado por mí.

A mis amigos de la universidad especialmente a los de mi apartamento que se convirtieron familia.

Agradezco profundamente a mis tutores que a pesar de ser personas muy ocupadas y con grandes responsabilidades siempre buscaban un tiempo para dedicármelo y hacer posible que llegase a la meta.

Agradezco a mis profesores, que me han guiado, apoyado y motivado a lo largo de mi formación académica.

Agradezco a mi mejor amigo y compañero de tesis, que ha estado presente en cada etapa de la universidad y me brindó su confianza y apoyo incondicional para emprender nuestro último desafío académico juntos.

Jeans Carlos

Agradecimientos

Agradezco profundamente a mis asesores de tesis, Madelin Haro Pérez y Víctor Alejandro Roque Domínguez, por su inestimable orientación, paciencia y apoyo durante todo el proceso de investigación y redacción de esta tesis.

Quiero expresar mi gratitud a mis amigos de Santa Clara que estudiaron conmigo o compartí momentos con ellos (José Manuel Cárdenas, entre otros que de una manera u otra aportó un granito de arena para la obtención de este logro personal). A mis compañeros de apartamento Alberto, Roy, Jorge Luis, Jeans Carlos, Yadian, Miguel y Brian ya que hemos recorrido toda esta aventura llena de alegrías, risas, decepciones, momentos difíciles, cuyo apoyo moral y académico fue fundamental para la realización de este trabajo.

Agradezco a mi familia, por su amor incondicional, su aliento constante y su fe en mis capacidades.

Agradezco a mi novia Estefany y mis suegros Ileana y Adalberto por su apoyo, ha sido mi fuerza motriz en los momentos más difíciles.

Agradezco a mi compañero de tesis Jeans Carlos Guzman Dorta que a su lado ha sido un honor trabajar en nuestro trabajo de diploma.

Finalmente, agradezco a los profesores que me ayudaron en mi desarrollo estudiantil y a la Universidad de Ciencias Informáticas por brindarme la oportunidad y los recursos para llevar a cabo esta investigación. Este trabajo no habría sido posible sin su apoyo.

Joan Carlos

Dedicatoria

Este trabajo es dedicado a nuestras familias fundamentalmente, ya que han sufrido y festejado los malos y buenos momentos por los que hemos atravesado durante nuestra vida como estudiantes.

Dedicamos este trabajo a todos nuestros profesores y compañeros que han sido parte de nuestro viaje académico. Su sabiduría y amistad han enriquecido nuestra vida de formas que no pueden expresar con palabras.

Este logro es tanto nuestro como suyo. A todos ustedes, va nuestro trabajo.

Resumen

La gestión educativa es un proceso estratégico integral que define objetivos, acciones y prioridades, comprometiendo a todos los actores institucionales según el tipo de educación. Consta de cuatro etapas: organización, planificación, ejecución y control. En las etapas de organización y planificación, se trazan las actividades, luego se ejecutan y evalúan a lo largo del curso. Un aspecto de este proceso es la evaluación integral del estudiante, que incorpora la caracterización, un registro detallado de las actividades del estudiante y sus logros a lo largo de su recorrido educativo.

El presente trabajo de tesis propone la implementación de un sistema de gestión de las actividades educativas de los estudiantes para su desarrollo integral en la Universidad de las Ciencias Informáticas. El sistema se desarrolló utilizando Python como lenguaje de programación, UML para el modelado, HTML5 para el marcado, y PyCharm versión 2022.2.3 como la herramienta principal de desarrollo. La implementación de este sistema facilita el crecimiento integral de los estudiantes al proporcionar un medio para registrar y controlar sus actividades. Dirigido a los actores responsables, a nivel más cercano al estudiante, de su seguimiento educativo, permite llevar un registro de las actividades que se realizan y de la participación y resultados de los estudiantes en estas, con vistas a su caracterización integral de fin de curso y previa a la ubicación laboral. Además, el sistema también permite el registro de premios y distinciones que los estudiantes han obtenido, así como la aplicación de sanciones por indisciplinas.

Palabras claves: caracterización, evaluación integral, gestión educativa, sistema de actividades educativas.

ABSTRACT

Educational management is an integral strategic process that defines objectives, actions, and priorities, involving all institutional actors according to the type of education. It consists of four stages: organization, planning, execution, and control. In the organization and planning stages, activities are outlined, then executed and evaluated throughout the course. One aspect of this process is the comprehensive evaluation of the student, which incorporates characterization, a

detailed record of the student's activities, and their achievements throughout their educational journey.

This thesis work proposes the implementation of a system for managing students' educational activities for their comprehensive development at the University of Informatics Sciences. The system was developed using Python as the programming language, UML for modeling, HTML5 for markup, and PyCharm version 2022.2.3 as the primary development tool. The implementation of this system facilitates the comprehensive growth of students by providing a means to record and control their activities. Aimed at those responsible, at the closest level to the student, for their educational follow-up, it allows keeping a record of the activities that are carried out and of the participation and results of the students in these, with a view to their comprehensive characterization at the end of the course and prior to job placement. In addition, the system also allows the registration of awards and distinctions that students have obtained, as well as the application of sanctions for indiscipline.

Keywords: characterization, comprehensive evaluation, educational management, system of educational activities.

Índice

Introducción	14
Capítulo 1. Los fundamentos y referentes teórico-metodológicos para el desarrollo de aplicaciones de gestión educativa en los años académicos de las universidades.	19
1.1 Proceso de organización y planificación de la gestión educativa.....	20
1.1.1 Los escenarios educativos en la educación superior	22
1.2 Proceso de ejecución y control de la gestión educativa.....	23
1.2.1 La evaluación integral	24
1.3 Estado del arte sobre los sistemas para la gestión educativa.....	25
1.4 Metodología, lenguaje y herramientas.....	29
1.4.1 Lenguaje de modelado	30
1.4.2 Lenguaje de programación	30
1.4.3 Lenguaje de marcado	31
1.4.4 Herramientas	31
Conclusiones del capítulo	32
Capítulo 2. Planificación y diseño del sistema para la gestión de las actividades educativas de los estudiantes para su desarrollo integral	34
2.1 Descripción de la propuesta de solución	34
2.2 Modelo del negocio:	34
2.3 Requisitos	35
2.4 Historias de usuarios.....	36
2.5 Estimación del tiempo por historias de usuario.....	42
2.6 Plan de duración de las iteraciones	43
2.7 Plan de entregas	44
2.8 Arquitectura de software.....	44
2.9.1 Patrones de diseño	45
2.9.2 Tarjetas CRC	46
2.10 Modelo de datos.....	49
Conclusiones del capítulo	50
Capítulo 3. Implementación y pruebas de la solución propuesta.	51
3.1 Estándares de codificación.....	51
3.2 Tareas de Ingeniería en la Metodología XP.....	52
3.3 Estrategia de pruebas	62
3.3.1 Pruebas unitarias.....	62
3.3.2 Pruebas de aceptación	65

3.4 Resultados de las pruebas	69
Conclusiones del capítulo	69
Conclusiones Generales	71
Referencias Bibliográficas	72

Índice de tablas

Tabla 1. Comparación de los sistemas homólogos	27
Tabla 2. Historia de usuario #1. Registrar la actividad en el sistema.....	38
Tabla 3. Historia de usuario #2. Registrar participación y resultado de una actividad	38
Tabla 4. Historia de usuario #3. Registrar sanción a un estudiante determinado	39
Tabla 5. Historia de usuario #4. Registrar premios y distinción a un estudiante determinado	39
Tabla 6. Historia de usuario #5. Generar caracterización del año de un estudiante determinado	40
Tabla 7. Historia de usuario #6. Gestión de los usuarios del sistema.....	41
Tabla 8. Historia de usuario #7. Generar caracterización integral del estudiante	41
Tabla 9. Estimación del tiempo por historias de usuario	42
Tabla 10. Plan de duración de iteraciones	43
Tabla 11. Plan de entregas.....	44
Tabla 12. Tarjeta CRC: AnnoAcademico	46
Tabla 13. Tarjeta CRC: ProfesorPrincipal	¡Error! Marcador no definido.
Tabla 14. Tarjeta CRC: Actividad.....	¡Error! Marcador no definido.
Tabla 15. Tarjeta CRC: Estudiante	48
Tabla 16. Tarjeta CRC: ProfesorGuia	48
Tabla 17. Tarjeta CRC: JefeDeGrupo	48
Tabla 18. Tarjeta CRC: Sancion	48
Tabla 19. Tarjeta CRC: Premio.....	49
Tabla 20. Tarea de Ingeniería: Insertar actividad	53
Tabla 21. Tarea de Ingeniería: Listar actividad	53
Tabla 22. Tarea de Ingeniería: Modificar actividad.....	53
Tabla 22. Tarea de Ingeniería: Eliminar actividad	54
Tabla 23. Tarea de Ingeniería: Registrar participación.....	54
Tabla 24. Tarea de Ingeniería: Registrar resultados	55
Tabla 25. Tarea de Ingeniería: Modificar registro de participación o resultados	55
Tabla 26. Tarea de Ingeniería: Eliminar registro de participación o resultados	55
Tabla 27. Tarea de Ingeniería: Insertar sanción.....	56
Tabla 28. Tarea de Ingeniería: Modificar sanción	56
Tabla 29. Tarea de Ingeniería: Listar sanciones	57
Tabla 30. Tarea de Ingeniería: Eliminar sanción.....	57
Tabla 31. Tarea de Ingeniería: Insertar premio o distinción	58
Tabla 32. Tarea de Ingeniería: Listar premio o distinción.....	58

Tabla 33. Tarea de Ingeniería: Modificar premios o distinciones.....	58
Tabla 34. Tarea de Ingeniería: Eliminar premios y distinciones	59
Tabla 35. Tarea de Ingeniería: Generar caracterización	59
Tabla 36. Tarea de Ingeniería: Modificar caracterización.....	60
Tabla 37. Tarea de Ingeniería: Insertar usuario	60
Tabla 38. Tarea de Ingeniería: Autenticar usuario	60
Tabla 39. Tarea de Ingeniería: Modificar usuario	61
Tabla 40. Tarea de Ingeniería: Eliminar usuario	61
Tabla 41. Tarea de Ingeniería: Generar caracterización integral del estudiante.....	61
Tabla 42. Tarea de Ingeniería: Modificar caracterización integral	62
Tabla 43. Caso de prueba de la tarea de ingeniería: Insertar la actividad	66

Índice de Figuras

Figura 1. Diagrama caso de uso del negocio.....	35
Figura 2. Funcionamiento del patrón Modelo-Vista-Plantilla.....	45
Figura 3. Modelo de datos.....	49
Figura 4: Código del método de prueba generar sanción	64
Figura 5: Grafo de flujo	65
Figura 6. Resultados de las pruebas	69

Introducción

La universidad cubana contemporánea se distingue por su enfoque en la formación humanista, moderna y universal, así como por su compromiso con la ciencia, la tecnología y la innovación. Esta institución está estrechamente vinculada con la sociedad, el sector productivo, los territorios y las comunidades, y trabaja para contribuir a la construcción de una nación socialista, independiente, soberana, democrática, próspera y sostenible. Uno de los mayores desafíos de la universidad cubana es fortalecer la formación integral de sus estudiantes para fomentar en ellos una mayor responsabilidad ética, económica, jurídica, social, cultural y ambiental. Esto implica formar profesionales no solo altamente capacitados en lo técnico, sino también dotados de cualidades personales y virtudes humanas que les permitan armonizar sus intereses individuales con los de la sociedad y participar activa y críticamente en su construcción (Ramos, Massip, Alfonso & Ronda, 2020).

El proceso de integralidad en la evaluación del estudiante universitario debe ofrecer elementos cualitativos y cuantitativos con datos suficientes sobre su desarrollo en los distintos años académicos. Asimismo, este proceso le brinda mecanismos para autorreconocer y reconocer por parte de otros, no solo las fallas, sino también las potencialidades y oportunidades que lo ayudan a superarlas (Ramos, Massip & Alfonso, 2017).

La evaluación integral del estudiante universitario adquiere un carácter institucional a partir del curso 2013-2014, cuando se le concede al colectivo de año académico el papel fundamental en su realización. Anteriormente, esta responsabilidad recaía en la Federación Estudiantil Universitaria (Ministerio de Educación Superior, 2018). De esta manera, se reconoce que el colectivo de cada año académico es el eslabón de base en el funcionamiento de la estructura organizativa de la universidad.

Es importante mencionar que, si bien existe un registro del decursar docente en el expediente y los sistemas informáticos utilizados para la gestión académica, el registro de la información del resto de las actividades con información detallada y específica sobre la vida estudiantil del individuo y sus necesidades formativas y educativas es escaso, insuficiente, o desorganizado o inexistente. Esto incluye la participación de los estudiantes en actividades como la guardia

estudiantil, cuartelaría, matutinos, actos políticos, desfiles, juegos deportivos, festivales culturales, jornada científica estudiantil, bastión entre otras.

En este punto se torna muy exigente el papel del profesor guía o del profesor principal del año, roles principales en el resultado docente-educativo del estudiante. El profesor principal elabora la estrategia educativa del año, conociendo en mayor o menor cuantía, a sus estudiantes, por tanto, el instrumento que debe soportar el desarrollo educativo del año, puede ser más o menos apropiado. Este, junto al profesor guía, llevan un registro, de acuerdo a sus iniciativas, de las actividades que realizan los estudiantes y con ello se conforman las caracterizaciones integrales de final de cada curso. Sin embargo, este control del año dependerá mucho de la comunicación de ambos roles y de su nivel de compromiso y responsabilidad con la tarea educativa asignada.

Tener el registro de las participaciones en las actividades, de los premios y/o medallas recibidos, de la contabilidad de los índices científicos y académicos, lograr hacer reportes, análisis y valoraciones del cumplimiento de los objetivos que se propone el año y que tributan a los cumplimientos de facultad y universidad, sería un elemento muy importante de control en el proceso docente-educativo que se lleva a nivel de año. Aunque han existido varios intentos que solucionan partes de esta problemática en la Universidad de las Ciencias Informáticas, no existe ningún instrumento que haya llegado a la actualidad, por no responder a las necesidades o adaptarse a la vida universitaria o lo haga de forma integral, manteniendo y facilitando la entrega pedagógica y el recorrido del estudiante desde su inicio en el centro de estudio.

Dada la problemática anterior se plantea el siguiente problema de investigación: ¿Cómo facilitar el registro documental de participación, resultados y reporte de las actividades educativas realizadas por los estudiantes durante toda la vida universitaria como respaldo a evidenciar su desarrollo integral? Se identifica como objeto de estudio: La gestión de actividades educativas en la Universidad de las Ciencias Informáticas.

Para solucionar el problema planteado se define como objetivo general: Desarrollar un sistema para la gestión de las actividades educativas en que participan los estudiantes que evidencie su desarrollo integral durante la vida estudiantil. Este objetivo se enmarca en el campo de acción: Sistemas informáticos para la gestión de las actividades educativas en la Universidad de las Ciencias Informáticas.

Con el fin de alcanzar el objetivo general se trazan los siguientes objetivos específicos:

1. Establecer los fundamentos y referentes teórico-metodológicos para el desarrollo de aplicaciones de gestión de actividades educativas en los años académicos de las universidades.
2. Diseñar una propuesta de solución que responda a la gestión de actividades educativas en los años académicos en las universidades cubanas.
3. Validar la solución para la gestión de las actividades educativas de los estudiantes para su desarrollo integral.

Con el propósito de dar cumplimiento a los objetivos establecidos, se elaboraron las siguientes **tareas de investigación**:

1. Fundamentación teórica sobre la gestión de actividades educativas en las universidades.
2. Análisis del estado del arte de sistemas informáticos para la gestión de actividades educativas en las universidades.
3. Documentación de la propuesta de solución haciendo uso de la modelación de su análisis y diseño ingenieril.
4. Desarrollar el sistema para la gestión de actividades educativas de los años académicos en la Universidad de las Ciencias Informáticas.
5. Validación del sistema para la gestión de las actividades educativas de los estudiantes para su desarrollo integral.

El desarrollo de la investigación se basará en la utilización de los siguientes métodos científicos categorizados en:

- Métodos teóricos:
 - Analítico-sintético: para la sistematización del conocimiento acerca de la gestión educativa de los años académicos en la UCI utilizando documentos normativos y enfoques de varios autores.
 - Método inductivo-deductivo: a emplearse en el análisis inductivo de los sistemas informáticos que utilizan, como funcionalidad base, la gestión educativa en diferentes universidades del mundo, para deducir posibles funcionalidades, comportamientos, y tecnologías usables en la propuesta de solución.

- Modelación: Se empleará para formular abstracciones con el propósito de interpretar la realidad, documentar la propuesta de solución sistema para la gestión de las actividades educativas de los estudiantes para su desarrollo integral.
- Métodos empíricos:
 - Observación: se utiliza como método para adquirir información sobre cómo se realizan los diferentes procesos de gestión educativa en la universidad para, posteriormente, utilizar información en la modelación y validación de la solución.
 - Entrevista: es el método para captar información que se utilizará para caracterizar la gestión educativa en la UCI y recopilar los requisitos, tanto funcionales como de comportamiento, del sistema.
 - Experimentación: método a ejecutar durante la validación del funcionamiento y cumplimiento de los objetivos de la investigación, a través de pruebas de software y otras técnicas.

La investigación obtendrá como posibles resultados:

1. La fundamentación teórico-metodológicos sobre el desarrollo de aplicaciones de gestión educativa en los años académicos de las universidades.
2. Un sistema web de gestión que permita facilitar el registro documental, actualización y reporte del diagnóstico de los datos personales y de la vida universitaria, durante los años académicos de los estudiantes en la UCI y la valoración integral por año y de carrera.
3. La documentación del sistema en gestión.

Con el desarrollo de un sistema para la gestión de las actividades educativas de los estudiantes para su desarrollo integral se esperan lograr beneficios desde los puntos de vista:

- Económicos: ya que se reducirán los costos asociados al uso de materiales de oficina y almacenamiento de documentos físicos. Generar ahorros en tiempo y recursos humanos al automatizar procesos y reducir errores.
- Sociales: favorecería la participación activa y protagónica de los estudiantes en su propio proceso formativo, al facilitar el acceso, seguimiento y retroalimentación de sus resultados académicos, extensionistas y sociopolíticos.

- Organizativos: mejora la eficiencia y eficacia en la gestión de la información integral de los estudiantes al permitir el registro organizado de la información, por año, estudiantes y tipos de actividades, el acceso a la información de los diferentes niveles de dirección asociados a los procesos docentes educativos y al control de procesos como la ejecución de la estrategia educativa y la caracterización e integralidad del estudiantado.
- Ambientales: contribuir a disminuir el impacto ambiental generado por el consumo de papel y otros materiales utilizados para documentar y gestionar la información integral de los estudiantes.

El informe de la tesis consta de portada, resumen, introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y anexos. Los capítulos se diseñan con una estructura informacional como se refiere a continuación:

- Capítulo I: “Los fundamentos y referentes teórico-metodológicos para el desarrollo de aplicaciones de gestión educativa en los años académicos de las universidades.” presenta un análisis conceptual, con el cual se examina las características de la gestión del proceso de integralidad. La información recopilada se utiliza para elaborar el estado del arte, teniendo en cuenta las soluciones informáticas disponibles para gestionar la integralidad de los estudiantes. Se evalúan las herramientas y tecnologías que deben integrarse al sistema general para el desarrollo de la solución óptima.
- Capítulo II: “Planificación y diseño del sistema de gestión de las actividades educativas de los estudiantes para su desarrollo integral”, se establecen las historias de usuarios, así como el diseño y la estructura de la aplicación que se crea. Se detallan los modelos de diseño utilizados y el enfoque arquitectónico adoptado.
- Capítulo III: “Implementación y pruebas de la solución propuesta”, incluye la estrategia de prueba, así como los casos y resultados obtenidos que permiten evaluar la satisfacción con la solución propuesta.

Capítulo 1. Los fundamentos y referentes teórico-metodológicos para el desarrollo de aplicaciones de gestión educativa en los años académicos de las universidades.

La gestión educativa es el proceso de desarrollo estratégico institucional de manera integral y coherente en el cual se definen: objetivos, acciones y prioridades que comprometen a todos los actores institucionales, de acuerdo al tipo de educación (Vidal, Durán & Pujal, 2008).

Responde a la formación integral de los futuros profesionales. La educación de la universidad es integral en la medida en que enfoque al estudiante como una totalidad y no lo considere únicamente por su potencial cognoscitivo o su capacidad técnica o profesional. El ámbito de la formación integral es el de una práctica educativa centrada en la persona humana y orientada a cualificar su socialización para que el estudiante pueda desarrollar su capacidad de servirse en forma autónoma del potencial de su espíritu en el marco de la sociedad en que vive y pueda comprometerse con sentido histórico en su transformación Fitch, 1997, citado por (Guzmán & García, 2017). Según Resolución 2/2018 “Reglamento de Trabajo Docente Metodológico de la Educación Superior” (Ministerio de Educación Superior, 2018) y ratificado en Resolución 47/2022 “Reglamento Organizativo del Proceso Docente y de dirección del Trabajo Docente y Metodológico para las carreras universitarias” (Ministerio de Educación Superior, 2022) en las universidades cubanas se desarrolla tanto curricular, con un enfoque docente educativo donde se integran las actividades académicas, laborales e investigativas; y extracurriculares, integrando la investigación científica extracurricular y la extensión universitaria.

Ambos términos incluyen en su desarrollo la gestión docente y la gestión extensionista. Para la gestión docente se apoya en el proceso docente educativo. Para Carlos Álvarez de Zayas (1998) “...el proceso docente educativo constituye el complemento del encargo social planteado a la escuela como problema a resolver; es el fin de egresar a los estudiantes en correspondencia con los intereses de nuestra sociedad socialista”. Autores más recientes como (Fuentes, Peña & Milán, 2010) lo conciben como el proceso que de modo consciente se desarrolla en las instituciones, a través de las relaciones de carácter social que se establecen entre sus participantes, con el propósito de educar, instruir y desarrollar a los futuros profesionales, sistematizando y recreando de forma planificada y organizada la cultura acumulada por la

humanidad y dando respuesta a las demandas de la sociedad. En fin, se considera el proceso docente educativo como un espacio de construcción de significados.

En consonancia con los términos tratados, corresponde establecer las actividades correspondientes a cada etapa de la gestión: organización, planificación, ejecución y control.

1.1 Proceso de organización y planificación de la gestión educativa

El Artículo 188 de la Sección Tercera, “Planificación, organización y regulación del proceso docente educativo”, en la Resolución 47/2022 “Reglamento Organizativo del Proceso Docente y de dirección del Trabajo Docente y Metodológico para las carreras universitarias” (Ministerio de Educación Superior, 2022), establece lo siguiente:

En la educación superior, la planificación y organización del proceso docente educativo constituyen dos de las funciones principales del trabajo metodológico. Se realizan antes del inicio del curso académico y de cada período docente en las instituciones de educación superior. Los planes de estudio de las carreras constituyen la base para la planificación y la organización del proceso docente educativo.

En este contexto de planificación y organización en la educación superior, se realiza un diagnóstico inicial antes del inicio del curso académico. Este diagnóstico, aplicado tanto al grupo en general como a cada estudiante en particular, es un componente esencial del proceso de formación de los estudiantes. Permite identificar las fortalezas y necesidades formativas del estudiante, en línea con los modos de actuación y las necesidades educativas propias de la universidad, facultad y territorio. A partir de los resultados obtenidos en este diagnóstico, se traza la estrategia educativa.

En este punto, es esencial definir la estrategia educativa como un plan de acciones diseñado para alcanzar objetivos comunes (Ministerio de Educación Superior, 2014). Esta estrategia, teniendo en cuenta los objetivos específicos de la carrera para la etapa de formación actual, así como los resultados del diagnóstico previo es elaborada por el Colectivo de Año Académico el cual está compuesto por los profesores guías, tutores, el grupo de profesores del año en curso, representantes estudiantiles y el profesor principal del año académico. Este último tiene la responsabilidad de liderar metodológicamente el trabajo docente educativo, quien en estrecha

coordinación con el profesor guía; debe mantener intercambios sistemáticos y participativos con los estudiantes

La estrategia educativa concibe acciones en tres dimensiones o esferas de influencias denominadas curriculares, extensión universitaria incluyendo otras actividades extracurriculares y socio-política y las articula entre sí para penetrar el tejido de toda la vida estudiantil universitaria (Alarcón Ortiz et al., 2019).

Dentro del marco de la estrategia educativa que se ha establecido, resulta crucial resaltar los aspectos fundamentales de las acciones educativas individuales, según lo establecido por el MES (Ministerio de Educación Superior, 2014):

1. Contiene acciones que reflejan tareas de impacto en la universidad en el territorio y en los marcos de la sociedad.
2. Incluye acciones concretas que realiza el estudiante en la residencia estudiantil como un espacio legítimo para su formación.
3. Compromiso individual de cada estudiante de los resultados académicos que debe alcanzar en cada asignatura en el semestre.
4. Compromiso de su participación en exámenes de premios.
5. Participación en las estrategias curriculares de preparación jurídica, preparación económica, preparación para la defensa, informatización, medio ambiente, Historia de Cuba e Idioma.
6. Participación en la investigación con resultados presentados en la jornada científica estudiantil.
7. Participación en las diferentes manifestaciones culturales y deportivas en los festivales de aficionados y en los juegos deportivos.
8. Participación como promotor cultural en el interior de la universidad y en la comunidad. (proyectos comunitarios)
9. Participación protagónica en las acciones de prevención y combate frontal contra las manifestaciones de fraude, indisciplinas, ilegalidades, corrupción y consumo de drogas.

10. Participación y cumplimiento de la guardia estudiantil.

11. Participación en las movilizaciones políticas convocadas por las organizaciones y por la institución.

12. Participación y resultados alcanzados en la preparación político ideológica que incluye el dominio de la actualidad nacional e internacional.

13. Responsabilidades individuales en la realización de las acciones que se convocan en el grupo de estudiantes.

1.1.1 Los escenarios educativos en la educación superior

La educación superior ha experimentado importantes cambios en los últimos años, que han implicado una transformación de los espacios donde se desarrolla el proceso de enseñanza aprendizaje. Estos espacios ya no se limitan al aula, sino que se amplían a otros contextos donde los estudiantes interactúan con la realidad social y económica del país.

Estos contextos se denominan escenarios educativos, y son aquellos espacios físicos donde transcurre la formación de los estudiantes. Los escenarios educativos se pueden clasificar en tres dimensiones: curricular, extensionista y sociopolítica.

La dimensión curricular se refiere a la estructura y contenido de los programas de estudio que se imparten en estas instituciones. Estos programas están diseñados para proporcionar a los estudiantes una formación académica sólida y relevante que les permita desempeñarse eficazmente en sus futuras carreras. Los espacios donde se imparten estos contenidos académicos son diversos y van más allá del aula tradicional. Pueden incluir talleres, empresas, museos y entidades culturales, entre otros. Estos espacios proporcionan a los estudiantes oportunidades para interactuar con la realidad social y económica del país, lo que enriquece su aprendizaje y les permite aplicar los conocimientos teóricos adquiridos en situaciones prácticas.

La dimensión extensionista abarca una serie de espacios y actividades que van más allá del currículo académico tradicional y que tienen un impacto significativo en la comunidad. Estos espacios incluyen las actividades en la residencia estudiantil, en el área deportiva, en el área cultural, y la participación en el trabajo socialmente útil. Cada uno de estos espacios ofrece a los estudiantes oportunidades únicas para aprender y crecer fuera del aula.

La dimensión sociopolítica en las universidades representa un pilar fundamental en el sistema de educación superior. Esta dimensión abarca una variedad de espacios donde los estudiantes tienen la oportunidad de participar activamente en actividades políticas. Estos espacios son vitales para el desarrollo de la ciudadanía activa y el compromiso con la revolución y el futuro de la nación.

Los escenarios educativos en la educación superior cubana son diversos y dinámicos, y responden a las necesidades y demandas de la realidad actual. Estos son una herramienta pedagógica que permite una formación integral, pertinente e innovadora de los estudiantes.

1.2 Proceso de ejecución y control de la gestión educativa

Las universidades tienen el encargo de desarrollar la labor formativa de profesionales revolucionarios comprometidos con nuestro proyecto social. La evaluación del estudiante universitario es una importante vía que contribuye a su proceso de formación y se debe realizar despojada de formalismo y con fuerte carácter motivador, que se caracterice por su transparencia. No menos importante la evaluación debe propiciar el compromiso y estímulo a reconocer las insuficiencias de los estudiantes y el deseo de resolverlas. Debe aspirar a que los estudiantes se propongan y comprometan con metas superiores, alcanzables con su esfuerzo y el del colectivo, que los conduzca a un crecimiento personal (Ministerio de Educación Superior, 2014).

En la Resolución 47/2022 “Reglamento Organizativo del Proceso Docente y de dirección del Trabajo Docente y Metodológico para las carreras universitarias” (Ministerio de Educación Superior, 2022), su Artículo 197 de la Sección Cuarta, “Del control al proceso docente educativo” establece a continuación el próximo párrafo: (...) es el medio fundamental para conocer la calidad de dicho proceso, evaluar sus resultados y dirigirlo hacia el cumplimiento de sus objetivos. Comprende, en lo fundamental:

- a) El control a la planificación y organización del proceso docente educativo en sus distintos niveles organizativos;
- b) el control a la calidad de la ejecución del proceso docente educativo; y
- c) el control a la preparación adquirida por los egresados.

A medida que avanza el año académico, las actividades se van ejecutando según lo planificado. Este proceso es esencial para asegurar que los estudiantes adquieran los conocimientos y habilidades necesarios en cada etapa de su formación integral.

Paralelamente a la ejecución de las actividades, se realiza un control de las mismas. Este control puede tomar varias formas, como la evaluación del rendimiento de los estudiantes en las tareas asignadas, la verificación de la asistencia y participación en las actividades extracurriculares, la obtención de premios, publicaciones o certificaciones, entre otros, en correspondencia con el tipo de actividad. Todos estos resultados deben tener evidencias, de forma tal que puedan ser exhibidos en los procesos de evaluación: seguimientos en la brigada del cumplimiento de la estrategia educativa, análisis en el colectivo de año, evaluaciones integrales al estudiante, entre otros.

1.2.1 La evaluación integral

La evaluación integral juega un papel crucial dentro del proceso de control. Esta se caracteriza por analizar con el estudiante los principales aspectos positivos e insuficiencias observados a lo largo del curso académico sobre la base de lo previsto en las diferentes dimensiones (curricular, extensionista y sociopolítica) de la estrategia educativa del año. Dicho análisis tiene como objetivo reforzar los aspectos positivos de la evaluación, discutir las principales insuficiencias de forma tal que permita que este ejercicio contribuya al proceso de formación de los estudiantes (Ramos, Massip, Alfonso & Ronda, 2020).

En el proceso de evaluación se utilizan diferentes fuentes de información y corresponde al profesor principal del año, de conjunto con su colectivo de profesores, elaborar la propuesta de evaluación. Por esa razón dicho proceso es institucional. El profesor principal del año puede delegar en el profesor guía la responsabilidad de la evaluación de su grupo de estudiantes (Ministerio de Educación Superior, 2014). Esta evaluación permite proponer vías y métodos efectivos para alcanzar las aspiraciones deseadas. Además, reconocen el carácter sistémico y continuo de este proceso. Este carácter sistemático se manifiesta en un proceso de evaluación integral que se lleva a cabo durante todo el curso académico, con revisiones mensuales, semestrales y anuales.

Para el desarrollo del proceso de evaluación integral es importante la caracterización del estudiante. Esta consiste en un análisis integral del desempeño de los estudiantes,

considerando sus aspectos curriculares, extensionistas y sociopolíticos. La caracterización abarca todas las actividades educativas en la que los estudiantes participan, así como sus resultados y reconocimientos, y también sus dificultades y contratiempos incluyendo alguna indisciplina que el estudiante pudiera cometer. La caracterización es esencial para la evaluación de los estudiantes, ya que permite detectar sus potencialidades y limitaciones, y orientar las acciones formativas que favorezcan su desarrollo integral.

1.3 Estado del arte sobre los sistemas para la gestión educativa

En el mundo, cada universidad, de acuerdo a su sistema social y sus normativas, también realizan gestión educativa e, igualmente, es una necesidad el apoyo en la tecnología para monitorearlo, apoyarlo y documentarlo. A continuación, se analizan algunos sistemas con el objetivo de analizar su estructura, funcionalidades y aplicabilidad a la solución del problema de esta investigación.

Ágora ERP, es un software de gestión líder para centros de formación, academias y escuelas. Este sistema proporciona una plataforma web integral para profesores, permitiéndoles llevar a cabo una variedad de tareas administrativas y académicas. Los profesores pueden pasar lista, subir documentos, calificar a los alumnos y consultar sus horarios.

Ágora ERP ofrece beneficios para los alumnos y los clientes. Pueden descargar calificaciones, facturas y otros documentos, y consultar sus horarios. Entre las funciones que ofrece este software se incluyen la gestión de alumnos, que permite controlar y gestionar la ficha de los alumnos, incluyendo sus datos personales y otros datos personalizables. También facilita la administración del personal docente y ayuda a gestionar y planificar cursos y grupos. Además, permite gestionar y verificar la impartición y asistencia a clase, así como facilitar la calificación y entrega de boletines de notas a los alumnos.

Additio App, es una plataforma de gestión escolar, evaluación y comunicación para profesores y centros educativos. Desde una perspectiva de gestión docente educativa ofrece una serie de herramientas que pueden ayudar a los profesores a mejorar la eficiencia y la eficacia en su trabajo pues brinda un cuaderno de notas digital personalizable que permite llevar un registro detallado del rendimiento académico de cada estudiante. Permite planificar y evaluar el currículo

académico. Este proceso lo posibilita utilizando varios métodos como la autoevaluación y la coevaluación, teniendo una carga objetiva y de justeza en el proceso.

Para los centros educativos, Additio App ofrece una plataforma de gestión administrativa, académica y de comunicación. Los administradores escolares pueden utilizarlo para gestionar todos los aspectos del centro educativo, desde la programación de clases hasta la comunicación con los padres. Además, ofrece informes y analíticas de aprendizaje que pueden ayudar a los administradores escolares a tomar decisiones informadas sobre cómo mejorar la calidad de la educación en su centro.

DATAFEU, es un software de gestión desarrollado para administrar los principales procesos de la Federación Estudiantil Universitaria (FEU) en la Universidad de las Ciencias Informáticas (UCI). Fue desplegado durante el curso 2011-2012 en la Facultad 3 y luego se extendió al resto de las facultades de la institución en el curso 2013-2014.

El sistema comprende un repositorio de datos que almacena y maneja la información relacionada con los estudiantes y procesos de enseñanza. Fue desarrollado sobre los marcos de trabajo Symfony y Ext JS, empleando MySQL como servidor de base de datos. Cuenta con cinco subsistemas que se integran entre sí y permiten el funcionamiento, integridad y persistencia de los datos que se introducen y procesan. Su problema principal radica en el acceso al almacenamiento de la información puesto que, aunque tiene una gestión por roles, no es descentralizada su introducción de datos a todos los actores del proceso educativo. Esta herramienta no está en uso desde hace varios años.

SIGENU, el Sistema de Gestión de la Nueva Universidad es un software desarrollado en la Universidad Tecnológica de la Habana “José Antonio Echeverría” (CUJAE) en el 2007. Se ha desarrollado con el fin de ser una herramienta que permita la gestión de toda la información académica vinculada con la educación superior.

El sistema SIGENU está compuesto por cuatro elementos fundamentales: Base de Datos, Servidor de Aplicaciones, Aplicación Cliente Secretaría y Aplicación Cliente Estudiante. La Base de Datos es el lugar en el cual se almacena toda la información registrada. El Servidor de Aplicaciones es el encargado de actuar como intermediario entre la Base de Datos y las aplicaciones clientes, permitiendo que la información registrada en la base sea visualizada y

actualizada a través de las aplicaciones que son manipuladas por el usuario. La Aplicación Cliente Secretaría es la aplicación que constituye el elemento que esencialmente permite la inserción y actualización de toda la información que se registre en el sistema.

En cuanto a las funcionalidades del sistema, SIGENU ofrece una gama de opciones para la gestión académica. Por ejemplo, permite llevar a cabo procedimientos relacionados con la matrícula, como matricular nuevo ingreso, actualizar datos de estudiantes matriculado, generar reportes de matrícula y realizar cierre de matrícula. También ofrece opciones para la gestión de estudiantes, como buscar estudiantes, modificar estudiante, matricular asignatura, evaluar estudiantes, generar reportes, agregar asignaturas electivas, otorgar bonificaciones y premios, eliminar estudiante y controlar baja de un estudiante.

Akademoss, utilizado en la Universidad de las Ciencias Informáticas (UCI), es una herramienta multiplataforma que contribuye solamente a la gestión académica de la institución, limitando la gestión educativa solo a este parámetro. Este sistema informático provee a estudiantes, profesores y personal administrativo una serie de funcionalidades que mejoran los procesos educativos.

Una de las principales ventajas del sistema Akademoss es su capacidad para centralizar toda la información relacionada con el ámbito académico. Los estudiantes pueden acceder a su plan de estudios, revisar sus calificaciones y consultar su promedio académico actual. Los profesores pueden llevar un registro detallado del desempeño académico de sus alumnos, cargar notas y evaluaciones. El sistema Akademoss cuenta con una interfaz intuitiva y fácil de usar, lo que facilita su adopción por parte de todos los usuarios.

En resumen, estos sistemas tienen potencialidades y desventajas a considerar en el análisis del estado del arte y en el aporte en cuanto a las funcionalidades y modo de hacer, para esta investigación. Este resumen aparece en la tabla 1.

Tabla 1. Comparación de los sistemas homólogos

Sistemas	Potencialidades	Desventajas
----------	-----------------	-------------

Ágora	La creación automática de documentación y la consolidación de la base de datos.	No cuenta con un registro documentado que refleje la participación de los estudiantes en las actividades. Aplicación de carácter propietario.
Aditio App	Lleva un registro del rendimiento académico de cada estudiante. Permite planificar y evaluar el currículo académico.	Se centra solo en la gestión académica. Aplicación de carácter propietario
DATAFEU	Recopila y gestiona datos sobre estudiantes y procesos de enseñanza, enfocándose en la caracterización e integralidad de los estudiantes	No está en uso desde hace varios años.
SIGENU	Implementación de módulos esenciales para la gestión de procesos de una institución universitaria	No guarda los datos relacionados a las dimensiones extensionista y sociopolítica.
AKADEMOS	Optimiza la planificación, coordinación y gestión de los procedimientos académicos.	No guarda los datos relacionados a las dimensiones extensionista y sociopolítica.

Como se puede apreciar, existe un esfuerzo mayor por la informatización de los procesos y la gestión más que de la visión educativa e integral del estudiante. En su mayoría, no se pudo acceder a los códigos fuentes por lo que no puede reutilizar partes de estos, las soluciones existentes no son adecuadas para las nuevas circunstancias. No tienen la habilidad de gestionar, controlar y mantener la información vinculada con la evaluación integral y el rendimiento de los estudiantes en los distintos ámbitos educativos y tienen una limitada adaptabilidad de roles según los permisos asignados por niveles. Por lo tanto, se confirma la necesidad de desarrollar un Sistema para la gestión docente educativa de los años académicos en la Universidad de las Ciencias Informáticas que sea accesible, flexible, colaborativo y seguro.

1.4 Metodología, lenguaje y herramientas

La **Metodología XP** o **Programación Extrema** es un enfoque ágil para el desarrollo de productos que se adapta a las necesidades y expectativas de los clientes, así como a los cambios que surgen durante el proceso. Esta metodología se distingue por la rapidez, la flexibilidad y la colaboración entre el cliente y el equipo de desarrollo. En este epígrafe, se describe esta metodología, se analizan sus principales características y se explica cómo se implementa en las diferentes etapas del desarrollo.

La **Metodología XP** es un conjunto de técnicas que dan agilidad y flexibilidad en la gestión de proyectos. Se centra en crear un producto según los requisitos exactos del cliente. De ahí, que le involucre al máximo durante el método de gestión del desarrollo del producto.

El uso de esta metodología supone, para muchos teóricos, una aproximación a la calidad óptima del producto. Pues durante el ciclo de vida del software, ocurren cambios naturales. Es más, cuantos más cambios, puede que más cerca estemos del mejor resultado que espera nuestro cliente. Por eso, este cambio constante en el proyecto se llega a considerar como favorable. Y si podemos aplicar una manera dinámica de gestionarlos, mejor. Esta forma es conocida como metodología XP.

Características de la Metodología XP (Asana, 2022):

- Comunicación constante entre el cliente y el equipo de desarrollo.
- Respuesta rápida a los cambios constantes.
- La planificación es abierta con un cronograma de actividades flexible
- Los requisitos del cliente y el trabajo del equipo del proyecto son los principales factores de éxito del mismo.

La Metodología XP, normalmente implica 4 fases o etapas del proceso de desarrollo que se repiten continuamente.

Etapas 1: Planificación. Se identificarán las historias de usuario, donde se establecen prioridades y se dividen en versiones más pequeñas. La planificación se revisará periódicamente. Cada iteración, que dura aproximadamente dos semanas, debe producir un software útil, funcional, listo para ser probado y lanzado.

Etapas 2: Diseño. En esta etapa, se trabajará con un código simple, haciendo lo mínimo necesario para que funcione. Además, para el diseño de software orientado a objetos, se crearán tarjetas CRC (Clase-Responsabilidad-Colaboración).

Etapas 3: Codificación colaborativa. La programación se realiza en parejas frente al mismo ordenador. De esta manera, se asegura que el código sea más universal y pueda ser entendido y trabajado por cualquier otro programador. Debe parecer que ha sido realizado por una sola persona, logrando así una programación organizada y planificada.

Etapas 4: Pruebas. Se deben realizar pruebas automáticas continuamente. Dado que normalmente se trata de proyectos a corto plazo, este testeo automatizado y constante es clave. Además, el propio cliente puede realizar pruebas, proponer nuevas pruebas y validar las versiones más pequeñas.

1.4.1 Lenguaje de modelado

El Lenguaje de Modelado Unificado (UML) es “un lenguaje estándar para escribir diseños de software. El UML puede usarse para visualizar, especificar, construir y documentar los artefactos de un sistema de software intensivo” (Pressman, 2010).

1.4.2 Lenguaje de programación

Python es un lenguaje de programación potente y fácil de aprender. Tiene Estructuras de datos de alto nivel y un enfoque simple pero efectivo para Programación orientada a objetos. La elegante sintaxis y la escritura dinámica de Python, junto con su naturaleza interpretada, lo convierten en un lenguaje ideal para el scripting y el rápido desarrollo de aplicaciones en muchas áreas en la mayoría de las plataformas (Python, 2023).

En el sitio web de Python están disponibles de forma gratuita en formato de código fuente o binario para todas las principales plataformas el intérprete de Python y su extensa biblioteca estándar. En el mismo sitio, también se puede encontrar distribuciones y enlaces a muchos módulos gratuitos de terceros para Python, programas, herramientas y documentación adicional.

1.4.3 Lenguaje de marcado

HTML5 es ampliamente utilizado como una palabra de moda para referirse a la web moderna. tecnologías, muchas de las cuales (aunque de ninguna manera todas) se desarrollan en el Grupo de Trabajo de Tecnología de Aplicaciones de Hipertexto Web (HTML, s. f.).

HTML5 convierte a HTML de un simple formato de marcado para estructurar documentos en una plataforma completa de desarrollo de aplicaciones. Entre otras características, HTML5 incluye nuevos elementos y API de JavaScript para mejorar el almacenamiento, la multimedia y el acceso al hardware (MDN Web Docs, 2022).

1.4.4 Herramientas

Visual Paradigm es un líder mundial de herramientas de diseño de software. Visual Paradigm es a la vez una herramienta de diseño y gestión para el desarrollo de sistemas de TI empresariales. Visual Paradigm está diseñado para ayudar a los equipos de desarrollo ágiles que desean una reducción de costos y una mejora de la calidad del software (Visual Paradigm, s. f.).

Visual Paradigm está disponible en múltiples plataformas, incluyendo Windows y Linux, y cuenta con un diseño centrado en casos de uso y enfocado al negocio, lo que permite generar software de alta calidad. Entre sus características destacan la generación de código para Java y la exportación como HTML, diagramas de procesos de negocio, modelado colaborativo con CVS y Subversión (control de versiones), diagramas de flujo de datos, soporte ORM para generar objetos Java desde bases de datos, generación de bases de datos desde diagramas de Entidad-Relación, ingeniería inversa desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación entre otras.

Pycharm es un entorno de desarrollo integrado (IDE) de Python dedicado que proporciona una amplia gama de herramientas esenciales para los desarrolladores de Python, estrechamente integradas para crear un entorno conveniente para el desarrollo productivo de ciencia de datos , web y Python (JetBrains, 2023)

En la edición Profesional de Pycharm, se pueden desarrollar aplicaciones Django, Flask y Pyramid. Además, es totalmente compatible con HTML (incluido HTML5), CSS, JavaScript y XML: estos idiomas se incluyen en el IDE a través de complementos y están activados de forma

predeterminada. La compatibilidad con otros lenguajes y marcos también se puede agregar a través de complementos. Todo lo que se hace en PyCharm, se hace dentro del contexto de un proyecto. Sirve como base para la asistencia de codificación, la refactorización masiva, la consistencia del estilo de codificación, etcétera (JetBrains, 2023).

Django es un marco de trabajo (framework) web de alto nivel que permite el desarrollo rápido de sitios web seguros y mantenibles. Desarrollado por programadores experimentados, Django se encarga de gran parte de las complicaciones del desarrollo web, por lo que puedes concentrarte en escribir tu aplicación sin necesidad de reinventar la rueda. Es gratuito y de código abierto, tiene una comunidad próspera y activa, una gran documentación y muchas opciones de soporte gratuito y de pago (García, 2015).

Django es un framework de desarrollo web que ayuda a los desarrolladores a escribir software completo, versátil, seguro, escalable, mantenible y portable. Django sigue la filosofía “Baterías incluidas” y provee casi todo lo que los desarrolladores quisieran que tenga “de fábrica”. Además, Django ayuda a los desarrolladores a evitar varios errores comunes de seguridad al proveer un framework que ha sido diseñado para “hacer lo correcto” para proteger el sitio web automáticamente. También es escalable y puede ser utilizado para construir casi cualquier tipo de sitio web. El código de Django está escrito usando principios y patrones de diseño para fomentar la creación de código mantenible y reutilizable. Además, Django está escrito en Python, lo que significa que puede ejecutarse en muchas plataformas.

Las herramientas que han sido mencionadas y definidas anteriormente son las que se emplearán en el desarrollo del sistema, ya que ofrecen las funcionalidades y los beneficios necesarios para lograr los objetivos propuestos.

Conclusiones del capítulo

- Se ha determinado que, para una evaluación efectiva de la formación integral de los estudiantes, es esencial tener en cuenta su participación y desempeño en diversas actividades a lo largo de su carrera académica. Esto incluye actividades docentes, extensionistas, políticas, científicas y productivas. Al hacerlo, se puede obtener una visión más completa del crecimiento y desarrollo del estudiante.

- Tras analizar el panorama actual, se determina que las soluciones existentes no abordan adecuadamente los nuevos desafíos. Por lo tanto, se confirma la necesidad de establecer un sistema para la gestión del proceso docente educativo de los años académicos en la Universidad de las Ciencias Informáticas.
- Se selecciona la metodología XP para desarrollar el proyecto, ya que ofrece adaptabilidad al cambio, la comunicación efectiva y la satisfacción al cliente. Se elige como lenguaje de desarrollo Python por su versatilidad y simplicidad, HTML5 como lenguaje de marcado por ser el estándar actual y más avanzado para crear páginas web dinámicas y finalmente el lenguaje UML para el modelado de las funcionalidades del sistema al ser este un lenguaje gráfico y universal que facilita la comprensión del software.

Capítulo 2. Planificación y diseño del sistema para la gestión de las actividades educativas de los estudiantes para su desarrollo integral

2.1 Descripción de la propuesta de solución

La propuesta de solución consiste en un sistema de gestión web que:

- Se desarrolle de forma incremental y adaptativa, incorporando nuevas funcionalidades según las necesidades de los usuarios.
- Contemple los componentes básicos para un sistema de gestión de actividades educativas para los profesores principales del año, profesores guías y estudiantes, tales como:
 - El registro de las actividades realizadas por los estudiantes, clasificadas por categorías y con sus respectivos datos.
 - El control de las asistencias a las actividades, así como de los premios, distinciones y sanciones que se otorguen o apliquen.
 - El sistema de visualización de la información referente a los procesos educativos.

2.2 Modelo del negocio:

Aunque no se concibe dentro de las fases de XP la modelación del negocio, para la realización de este trabajo de diploma se considera oportuno realizarla a través del modelo de caso de uso del negocio.

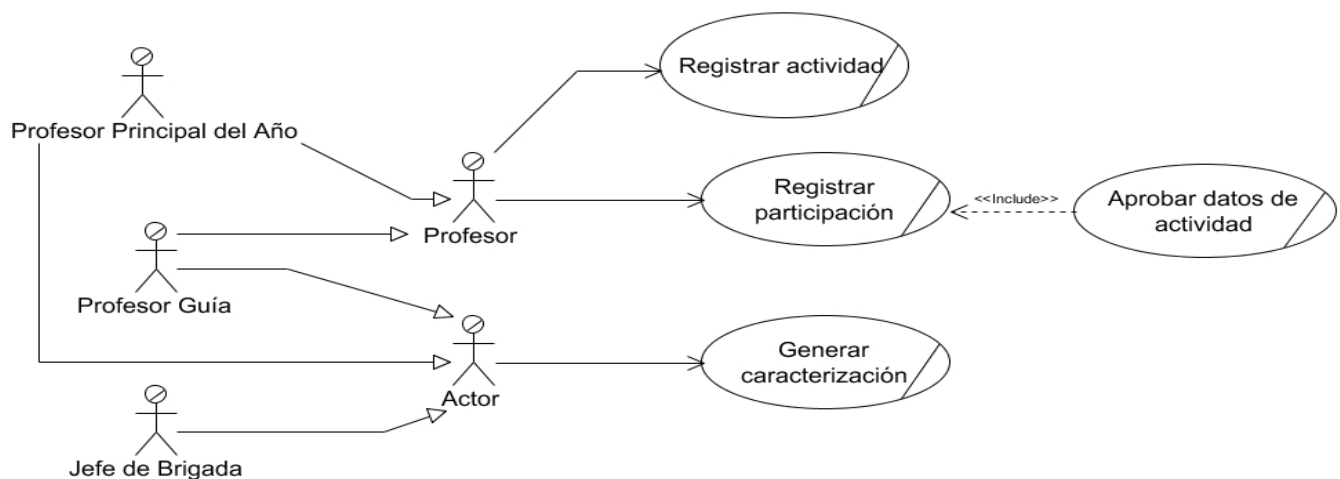


Figura 1. Diagrama caso de uso del negocio

En este modelo, los actores son:

- **Profesor principal:** responsable de planificar, coordinar y evaluar las actividades educativas de los estudiantes. Registra y analiza los datos de asistencia y participación de los estudiantes y elabora la caracterización de los mismos.
- **Profesor guía:** encargado de acompañar, orientar y apoyar a los estudiantes de su grupo en su proceso formativo. Puede proponer y ejecutar actividades que hayan sido aprobadas por el profesor principal, así como registrar la participación y los resultados de las mismas. También contribuye a la elaboración de las caracterizaciones de los estudiantes de su grupo.
- **Jefe de brigada:** puede participar en la elaboración de las caracterizaciones de los estudiantes de su grupo.

Las funcionalidades principales que se llevan a cabo en este modelo son:

- **Registrar actividad:** Esta actividad implica la planificación y documentación de las actividades educativas que se llevarán a cabo.
- **Generar caracterización:** un registro detallado de las actividades que participó el estudiante, sus logros e indisciplinas a lo largo de su recorrido educativo.
- **Registrar participación:** Esto implica llevar un registro de la asistencia y participación de los estudiantes en las actividades educativas.
- **Aprobar datos de la actividad:** Esto implica la revisión y aprobación de los datos generados a partir de las actividades educativas.

2.3 Requisitos

Los requisitos para un sistema son descripciones de lo que el sistema debe hacer: el servicio que ofrece y las restricciones en su operación. Tales requisitos reflejan las necesidades de los clientes por un sistema que atienda cierto propósito (Sommerville, 2011). Para el desarrollo de

este trabajo de diploma es factible identificar y representar tanto los requisitos funcionales como los no funcionales, aunque estos no se conciben dentro de las fases de la metodología XP.

Los requisitos funcionales son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requisitos funcionales también pueden declarar explícitamente lo que el sistema no debe hacer (Sommerville, 2011).

- RF1. Registrar actividad del estudiante
- RF2. Listar actividad
- RF3. Modificar actividad
- RF4. Eliminar actividad
- RF5. Registrar participación del estudiante en la actividad
- RF6. Modificar la participación del estudiante en la actividad
- RF7. Registrar distinciones y premios
- RF8. Modificar distinciones y premios
- RF9. Eliminar distinciones y premios
- RF10. Listar distinciones y premios
- RF11. Registrar sanciones
- RF12. Modificar sanciones
- RF13. Eliminar sanciones
- RF14. Listar sanciones
- RF15. Generar caracterización del estudiante
- RF16. Modificar caracterización
- RF17. Elaborar plantilla de integralidad del estudiante
- RF18. Autenticar usuario
- RF19. Crear usuario
- RF20. Modificar usuario
- RF21. Eliminar usuario
- RF22. Buscar (estudiante, actividad, premio, distinción, sanción)

Los requisitos no funcionales son limitaciones sobre servicios o funciones que ofrece el sistema. Incluyen restricciones tanto de temporización y del proceso de desarrollo, como impuestas por

los estándares. Los requerimientos no funcionales se suelen aplicar al sistema como un todo, más que a características o a servicios individuales del sistema (Sommerville, 2011). A continuación, son mencionados los requisitos no funcionales para el desarrollo del sistema:

- Usabilidad:
 - El sistema debe ser fácil de usar y comprender para el usuario.
 - La disposición y el diseño de los elementos del sistema en la pantalla deben ser consistentes; por lo tanto, el diseño debe ser adaptable a diferentes tamaños de pantalla (Responsive).
- Seguridad:
 - La seguridad del sistema se basa en diferentes niveles de acceso, determinados por roles asignados a los usuarios que interactúan con el sistema. Se basa en niveles de acceso de acuerdo al rol del usuario, que identifica la información que puede ser editada, consultada.
 - La información procesada por el sistema debe estar resguardada contra accesos no autorizados y divulgación.
- Rendimiento:
 - Los tiempos de respuesta a las solicitudes no deben exceder los dos segundos.

2.4 Historias de usuarios

La metodología XP propone dividir el proyecto en pequeñas partes manejables, cada una representada por una historia de usuario. Las historias de usuario se usan, en el contexto de la ingeniería de requisitos ágil, como una herramienta de comunicación que combina las fortalezas de ambos medios: escrito y verbal. Describen, en una o dos frases, una funcionalidad de software desde el punto de vista del usuario, con el lenguaje que éste emplearía (Menzinsky et al., 2022). A continuación, son mencionadas las historias de usuarios que fueron identificadas:

- | | |
|------|--|
| HU.1 | Registrar la actividad en el sistema. |
| HU.2 | Registrar participación, evidencia y resultado de una actividad. |
| HU.3 | Registrar sanción a un estudiante determinado. |
| HU.4 | Registrar premio y distinción a un estudiante determinado. |
| HU.5 | Generar caracterización del año de un estudiante determinado. |
| HU.6 | Gestión de los usuarios del sistema. |

HU.7 Generar caracterización integral del estudiante.

Tabla 2. Historia de usuario #1. Registrar la actividad en el sistema

Historia de usuario	
Identificador: HU.1	Nombre: Registrar la actividad en el sistema.
Usuario: Profesor principal	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1.60	Iteración asignada: 1
Programadores: - Joan Carlos Galván Díaz - Jeans Carlos Guzman Dorta	
Descripción: Yo, profesor principal quiero registrar una actividad en el sistema para poder indicar los datos que deben llenarse al registrar una participación.	
Criterio de aceptación: - Debe haber un formulario que posea los campos: nombre, tipo de actividad, fecha de inicio, fecha de fin, clasificación, que el usuario puede ir completando a su conveniencia. - Debe poderse seleccionar los tipos de actividad. - Debe poderse seleccionar si la actividad es masiva, selectiva o individual. - Debe poderse modificar o eliminar una actividad insertada en la lista de actividades existentes.	

Tabla 3. Historia de usuario #2. Registrar participación y resultado de una actividad

Historia de usuario	
Identificador: HU.2	Nombre: Registrar participación y resultado de una actividad.
Usuario: Profesor principal	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1.40	Iteración asignada: 1
Programadores: Joan Carlos Galván Díaz	

• Jeans Carlos Guzman Dorta
Descripción: Yo, profesor principal quiero registrar la participación y el resultado de los estudiantes en una actividad ya registrada para registrar su desarrollo educativo.
Criterio de aceptación: • Debe de aparecer en las actividades insertadas la opción para seleccionar si el estudiante participó o no en una actividad. • En dependencia del tipo de actividad se debe poder seleccionar si el estudiante tuvo resultados o no.

Tabla 4. Historia de usuario #3. Registrar sanción a un estudiante determinado

Historia de usuario	
Identificador: HU.3	Nombre: Registrar sanción a un estudiante determinado.
Usuario: Profesor Principal	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1.20	Iteración asignada: 2
Programadores: • Joan Carlos Galván Díaz • Jeans Carlos Guzman Dorta	
Descripción: Yo, profesor principal quiero registrar una sanción en el sistema para poder evidenciar la falta, la clasificación y la sanción a una indisciplina cometida por un estudiante.	
Criterio de aceptación: • Debe haber un formulario que posea los campos: nombre, clasificación, fecha y número de resolución. • Debe poderse modificar o eliminar una sanción insertada en la lista de sanciones existentes.	

Tabla 5. Historia de usuario #4. Registrar premio y distinción a un estudiante determinado

Historia de usuario

N Identificador: HU.4	Nombre: Registrar premio y distinción a un estudiante determinado.
Usuario: Profesor Principal	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1.20	Iteración asignada: 2
Programadores: - Joan Carlos Galván Díaz - Jeans Carlos Guzman Dorta	
Descripción: Yo, profesor principal quiero registrar un premio o una distinción adquirida por un estudiante con el fin de tener evidencia del mismo.	
Criterio de aceptación: - Debe haber un formulario que posea los campos: nombre, tipo, fecha. - Debe poderse modificar o eliminar un premio o distinción insertado en la lista de premios o distinciones existentes.	

Tabla 6. Historia de usuario #5. Generar caracterización del año de un estudiante determinado

Historia de usuario	
Identificador: HU.5	Nombre: Generar caracterización del año de un estudiante determinado.
Usuario: Profesor Principal	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 3
Programadores: - Joan Carlos Galván Díaz - Jeans Carlos Guzman Dorta	
Descripción: Yo, profesor principal quiero generar la caracterización del año de un estudiante determinado para guardarla y entregarla en secretaria para registrar en el expediente del estudiante.	
Criterio de aceptación:	

- Se debe generar la caracterización donde aparecen las actividades que el estudiante ha participado, premios, distinciones y sanciones que pudo haber obtenido durante un año académico.
- Se puede modificar para registrar otros elementos no recogidos por el sistema como su decursar académico, misiones, entre otros.

Tabla 7. Historia de usuario #6. Gestión de los usuarios del sistema

Historia de usuario	
Identificador: HU.6	Nombre: Gestión de los usuarios del sistema.
Usuario: Administrador	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 3
Programadores: • Joan Carlos Galván Díaz • Jeans Carlos Guzman Dorta	
Descripción: Administrador es el encargado de gestionar los usuarios del sistema.	
Criterio de aceptación: • Se deben poder crear usuarios asignando un usuario y una contraseña. • Se debe poder asignarles permisos a los usuarios. • Se debe poder modificar un usuario. • Se debe poder eliminar un usuario.	

Tabla 8. Historia de usuario #7. Generar caracterización integral del estudiante

Historia de usuario	
Identificador: HU.7	Nombre: Generar caracterización integral del estudiante.
Usuario: Profesor Principal	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 3
Programadores:	

- Joan Carlos Galván Díaz
- Jeans Carlos Guzman Dorta

Descripción: Yo, profesor principal quiero generar la caracterización integral.

Criterio de aceptación:

- Se debe generar la caracterización donde aparecen todas las actividades que el estudiante ha participado, premios, distinciones y sanciones que pudo haber obtenido durante toda su vida estudiantil.
- Se puede modificar las caracterizaciones.

2.5 Estimación del tiempo por historias de usuario

La metodología XP, o Programación Extrema, propone un enfoque de estimación de esfuerzo basado en las historias de usuario. Este enfoque facilita la determinación del tiempo requerido para el desarrollo del software, permitiendo una gestión más eficaz y precisa.

En XP, los programadores utilizan puntos como medida para establecer las estimaciones de esfuerzo asociadas a la implementación de las historias. Cada punto equivale a una semana ideal de programación. Por lo general, las historias se valoran entre 1 y 3 puntos.

Tabla 9. Estimación del tiempo por historias de usuario

Historias de usuarios	Puntos estimados
Registrar la actividad	1.60
Registrar participación y resultado de una actividad	1.40
Registrar sanción a un estudiante determinado	1.40
Registrar premios y distinción a un estudiante determinado	1.60
Generar caracterización del año de un estudiante determinado.	1
Gestión de los usuarios del sistema.	1

Generar caracterización integral del estudiante.	1
--	---

2.6 Plan de duración de las iteraciones

Una vez que se han definido las historias de usuario y se ha realizado la estimación de esfuerzos, el siguiente paso es planificar la fase de implementación del sistema. Esta planificación consiste en determinar la duración, el orden y la prioridad de cada iteración, que es un ciclo corto de desarrollo que entrega una parte funcional del software. La planificación de la implementación del sistema tiene como objetivo optimizar el proceso de desarrollo, asegurando que se entreguen las funcionalidades más importantes y urgentes en primer lugar, y que se cumplan los plazos establecidos.

Tabla 10. Plan de duración de iteraciones

Iteración	Historias de usuarios	Puntos estimados
Iteración 1	Registrar la actividad	3
	Registrar participación y resultado de una actividad	
Iteración 2	Registrar sanción a un estudiante determinado	3
	Registrar premios y distinción a un estudiante determinado	
Iteración 3	Generar caracterización del año de un estudiante determinado.	3
	Gestión de los usuarios del sistema.	
	Generar caracterización integral del estudiante.	

Tabla 10. Plan de duración de iteraciones

2.7 Plan de entregas

El Plan de Entrega en XP se basa en la idea de realizar entregas incrementales y frecuentes de software funcional. El plan se divide en iteraciones de no más de tres semanas, lo que permite al equipo desarrollar, probar y entregar nuevas funcionalidades de manera rápida y constante.

Tabla 11. Plan de entregas

Iteración	Fecha de inicio	Fecha de fin
1	18/9/23	6/10/23
2	9/10/23	27/10/23
3	30/10/23	17/11/23

2.8 Arquitectura de software

La arquitectura de software forma la columna vertebral para construir un sistema de software, es en gran medida responsable de permitir o no ciertos atributos de calidad del sistema entre los que se destacan la confiabilidad y el rendimiento del software. La manera en que se estructura un sistema permitirá o impedirá que se satisfagan las necesidades del cliente. Además, la arquitectura de software juega un papel fundamental para guiar el desarrollo. Una de las múltiples estructuras que la componen se enfoca en partir el sistema en componentes que serán desarrollados por individuos o grupos de individuos (González, 2020).

Para aplicar una arquitectura de software de forma efectiva, se utiliza el patrón de diseño Model-Template-View (MTV), que facilita la separación de responsabilidades entre las capas de una solución tecnológica. Este patrón de diseño permite que cada componente tenga una función específica y bien definida, lo que mejora a modularidad, la reusabilidad y la mantenibilidad del código. El patrón de diseño MTV se compone de los siguientes elementos (Mozilla, 2023):

- Vista (View): Una vista es una función de gestión de peticiones que recibe peticiones HTTP y devuelve respuestas HTTP. Las vistas acceden a los datos que necesitan para satisfacer las peticiones por medio de modelos, y delegan el formato de la respuesta a las plantillas ("templates").
- Modelos (Models): Los Modelos son objetos de Python que definen la estructura de los datos de una aplicación y proporcionan mecanismos para gestionar (añadir, modificar y borrar) y consultar registros en la base de datos.

- Plantillas (Templates): una plantilla (template) es un fichero de texto que define la estructura o diagrama de otro fichero (tal como una página HTML), con marcadores de posición que se utilizan para representar el contenido real. Una vista puede crear dinámicamente una página usando una plantilla, rellenándola con datos de un modelo. Una plantilla se puede usar para definir la estructura de cualquier tipo de fichero; ¡no tiene porqué ser HTML!

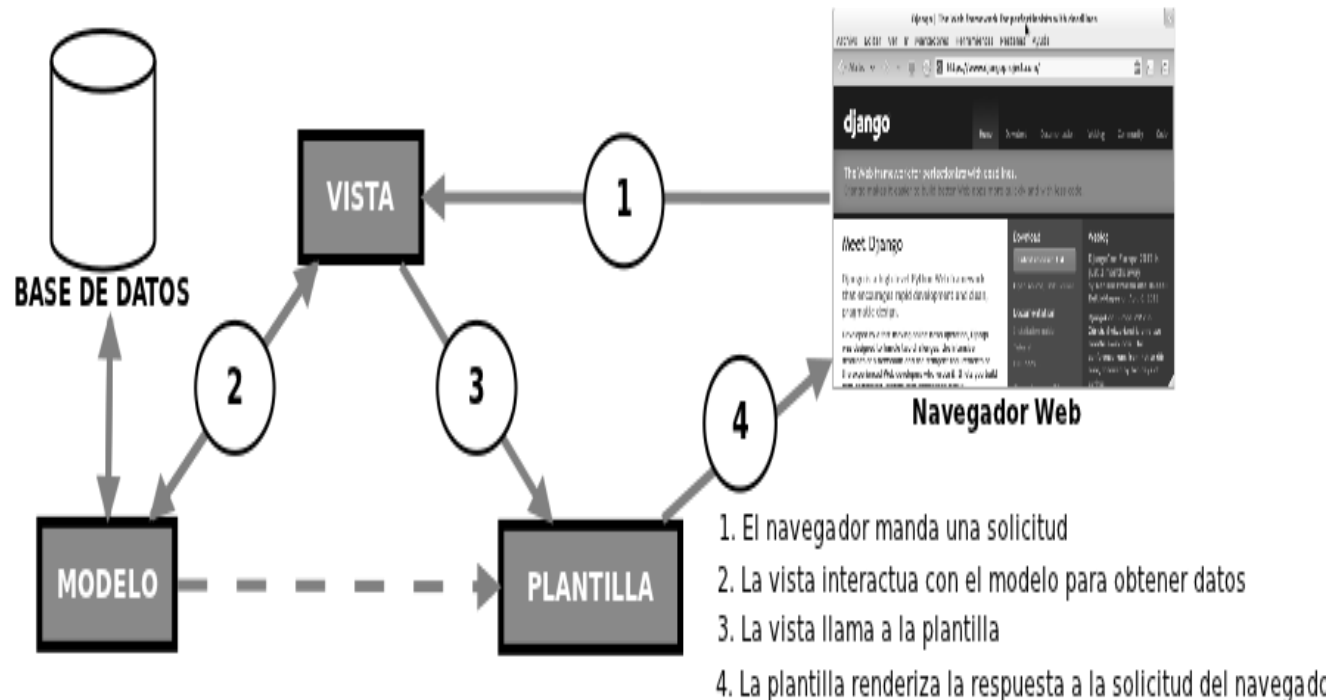


Figura 2. Funcionamiento del patrón Modelo-Vista-Plantilla

2.9.1 Patrones de diseño

Los patrones de diseño son una herramienta esencial en el desarrollo de software, proporcionando soluciones probadas y eficientes a problemas comunes que surgen durante el proceso de diseño (García, 2023). Los patrones de diseño promueven la reutilización de código y pueden hacer que el código sea más fácil de entender y mantener, ya que proporcionan abstracciones de alto nivel para problemas de diseño comunes.

Entre los diversos tipos de patrones de diseño que existen, se encuentra el patrón GRASP, que se caracteriza por ser un conjunto de principios generales para asignar responsabilidades a las clases y objetos en el desarrollo de software orientado a objetos. A continuación, se caracteriza dicho patrón:

Los principios GRASP que se aplicaron son los siguientes:

- Alta Cohesión: Este principio se refleja en la mayoría de las clases del diseño, ya que cada una se encarga de implementar solo las funcionalidades que le son propias.
- Bajo Acoplamiento: Este principio se evidencia en que cada clase interactúa con un número limitado de otras clases.
- Creador: Las clases encargadas de crear objetos poseen la información necesaria para su construcción.
- Experto: Este principio se cumple al mantener el encapsulamiento, permitiendo que los objetos utilicen su propia información para realizar sus tareas. El comportamiento se distribuye entre las clases que poseen la información necesaria, lo que facilita su comprensión y mantenimiento.

Además, se utilizaron los siguientes patrones GoF:

- Controlador Frontal (Front Controller): Django tiene una implementación de Controlador Frontal que dirige las solicitudes a métodos o clases, que en la práctica son controladores de páginas. Antes del despacho, la solicitud es procesada por varios filtros (middlewares).
- Decorador (Decorator): Este patrón permite agregar responsabilidades adicionales a un objeto de manera dinámica, ofreciendo una alternativa flexible a la especialización a través de la herencia cuando se trata de agregar funcionalidades.

2.9.2 Tarjetas CRC

Las tarjetas CRC (Clases-Responsabilidades-Colaboradores) son una herramienta esencial que permite a los programadores enfocarse en el desarrollo orientado a objetos, facilitando la transición desde la programación procedimental clásica y ayudando a evitar sus malos hábitos. Estas tarjetas representan de manera efectiva los objetos dentro del sistema. En cada tarjeta CRC, la clase a la que pertenece un objeto se anota en la parte superior, mientras que las responsabilidades y colaboraciones asociadas al objeto se detallan en la parte inferior. De esta manera, las tarjetas CRC proporcionan una representación visual clara y concisa de los objetos y sus interacciones dentro del sistema.

Tabla 12. Tarjeta CRC: AnnoAcademico

AnnoAcademico	
Responsabilidades	Colaboradores
• Contener grupos. • Es representado por un profesor principal.	• ProfesorPrincipal • Grupo

Tabla 13. Tarjeta CRC: ProfesorPrincipal

ProfesorPrincipal	
Responsabilidades	Colaboradores
• Aprobar y registrar en el sistema las diferentes actividades que participan los estudiantes. • Registrar la participación de los estudiantes en las actividades. • Insertar, modificar y eliminar premios obtenidos por los estudiantes. • Insertar, modificar y eliminar sanciones a un estudiante por una indisciplina cometida.	• AnnoAcademico • Actividad • Premio • Sancion

Tabla 14. Tarjeta CRC: Actividad

Actividad	
Responsabilidades	Colaboradores
• Nombre • Tipo • Fecha de inicio • Fecha de fin	• ProfesorPrincipal • ProfesorGuía • JefeDeGrupo

• Puede ser aprobada o eliminada por el profesor principal	
--	--

Tabla 15. Tarjeta CRC: Estudiante

Estudiante	
Responsabilidades	Colaboradores
• Participar en actividades • Obtener premios • Cometer indisciplina y recibir sanciones	• Actividad • Premio • Sancion

Tabla 16. Tarjeta CRC: ProfesorGuia

ProfesorGuia	
Responsabilidades	Colaboradores
• Registrar y administrar actividades con aprobación del Profesor Principal	• ProfesorPrincipal

Tabla 17. Tarjeta CRC: JefeDeGrupo

JefeDeGrupo	
Responsabilidades	Colaboradores
• Registrar y administrar actividades con aprobación del Profesor Principal	• ProfesorPrincipal

Tabla 18. Tarjeta CRC: Sancion

Sancion

Conclusiones del capítulo

- El sistema debe responder a 7 funcionalidades, las cuales se describen en igual número de historias de usuario.
- Son necesarias la realización de 3 iteraciones, con una duración total de 9 semanas.
- Se concluye estructurar el sistema con la arquitectura en 3 capas utilizando como patrón de diseño el Modelo-Vista-Plantilla.

.

Capítulo 3. Implementación y pruebas de la solución propuesta.

Siguiendo la metodología XP y luego de planificar y analizar el sistema, se procede a la implementación. Está guiada por las tareas de ingeniería. Se efectúan pruebas unitarias para verificar la funcionalidad adecuada de la aplicación y pruebas de aceptación para asegurar que el producto final cumpla con las expectativas del cliente.

3.1 Estándares de codificación

Guido van Rossum, el creador del lenguaje de programación Python, ha enfatizado que el código se lee mucho más de lo que se escribe. Por lo tanto, es esencial tener directrices claras para garantizar la legibilidad y consistencia del código.

Para este propósito, se utiliza la Guía de Estilo para el Código Python, también conocida como PEP 8. Esta guía ofrece una serie de convenciones para escribir código de manera clara y legible. Algunas de las convenciones más destacadas son:

- **Nombres y convenciones:** Los nombres de los módulos, clases, funciones, variables y constantes deben seguir ciertas reglas de estilo, como usar minúsculas, mayúsculas, guiones bajos o palabras completas, para mejorar la legibilidad y evitar conflictos. Los nombres que empiezan o terminan con dos guiones bajos se reservan para usos especiales, como los métodos mágicos o las variables dunder.
- **Indentación y líneas en blanco:** Se debe usar cuatro espacios por nivel de indentación, y evitar el uso de tabulaciones. Las líneas no deben exceder los 79 caracteres, y se pueden romper usando paréntesis, corchetes o llaves implícitas, o usando una barra invertida. Se deben usar líneas en blanco para separar las definiciones de funciones y clases, y para indicar secciones lógicas dentro de las funciones. Los comentarios deben ser oraciones completas y empezar con una mayúscula.
- **Importaciones:** Las importaciones deben estar en líneas separadas, y preferiblemente al principio del archivo, después de los comentarios y las cadenas de documentación. Se deben usar importaciones absolutas en lugar de relativas, y evitar las importaciones comodín (`from <module> import *`). Se debe usar el atributo `all` para declarar los nombres públicos de un módulo.

- **Espacios en blanco:** Se deben evitar los espacios en blanco innecesarios, como alrededor de los paréntesis, corchetes o llaves, o antes de las comas, punto y coma o dos puntos. Se debe usar un espacio alrededor de los operadores y después de las comas, excepto en los casos donde se use una indentación colgante. Se debe usar un espacio después del símbolo # en los comentarios.
- **Comentarios:** Los comentarios deben ser claros y fáciles de entender para otros hablantes del idioma que se esté usando. Se deben usar cadenas de documentación (docstrings) para documentar los módulos, funciones, clases y métodos públicos, siguiendo las convenciones de PEP 257. Se deben usar comentarios de bloque para explicar el código que no es obvio, y comentarios en línea para aclarar detalles específicos.
- **Anotaciones:** Las anotaciones de funciones y variables deben usar la sintaxis de PEP 484, y tener un espacio después de los dos puntos. Las anotaciones de funciones deben indicar los tipos de los argumentos y el valor de retorno, y las anotaciones de variables deben indicar el tipo de la variable. Las anotaciones son opcionales y no afectan al comportamiento del intérprete, pero pueden ser usadas por herramientas externas como los verificadores de tipos.

3.2 Tareas de Ingeniería en la Metodología XP

La metodología XP (Extreme Programming) tiene como objetivo lograr una comunicación clara con el cliente. Una de las formas de hacerlo es transformando las historias de usuario en tareas de ingeniería, que son más fáciles de entender y seguir. Cada tarea de ingeniería se define por los siguientes campos:

- Nombre de la tarea: Identificador único de la tarea.
- HU: Corresponde a la historia de usuario a la que pertenece la tarea.
- Puntos estimados: Tiempo estimado en días para el desarrollo de la tarea.
- Fecha de inicio: Fecha en que se inicia el desarrollo de la tarea.
- Fecha de fin: Fecha en que se concluye el desarrollo de la tarea.
- Descripción: Breve descripción de la tarea.

Estos campos permiten un seguimiento detallado de cada tarea de ingeniería, asegurando que se cumplan las especificaciones solicitadas por el cliente. A continuación, se presentan las tareas de ingeniería:

Tabla 20. Tarea de Ingeniería: Insertar actividad

Tarea de Ingeniería	
Nombre de la tarea: Insertar actividad	
HU: Registrar la actividad en el sistema	Puntos estimados: 0.6
Fecha de inicio: 18/9/23	Fecha de fin: 20/9/23
Descripción: Un formulario se muestra a los usuarios, solicitando detalles específicos para registrar una nueva actividad. Los campos a completar incluyen el nombre de la actividad, su clasificación, la fecha de inicio y de finalización, así como el tipo de actividad, que puede ser individual o grupal. El formulario tiene la opción de cancelar y la opción de aceptar una vez llenado todos los campos. En caso del usuario dar clic en aceptar, la información de la actividad se almacena en la base de datos del sistema. Esta actividad registrada se vuelve accesible para los usuarios.	

Tabla 21. Tarea de Ingeniería: Listar actividad

Tarea de Ingeniería	
Nombre de la tarea: Listar actividad	
HU: Registrar la actividad	Puntos estimados: 0.4
Fecha de inicio: 21/9/23	Fecha de fin: 22/9/23
Descripción: A través de una interfaz de usuario diseñada de manera intuitiva, se muestra al usuario una lista organizada de actividades. Esta lista visualiza detalles de cada actividad, incluyendo el nombre de la actividad, su clasificación, la fecha de inicio y de finalización, así como el tipo de actividad. Además, la interfaz proporciona opciones para la gestión de actividades. Estas opciones permiten al usuario insertar nuevas actividades, modificar actividades existentes y eliminar actividades según sea necesario. Esta funcionalidad integral facilita la administración eficiente de las actividades por parte del usuario.	

Tabla 22. Tarea de Ingeniería: Modificar actividad

Tarea de Ingeniería	
Nombre de la tarea: Modificar actividad	
HU: Registrar la actividad	Puntos estimados: 0.4
Fecha de inicio: 25/9/23	Fecha de fin: 26/9/23
Descripción: El usuario tiene la opción de seleccionar de la lista de actividades una actividad específica que desea modificar. Al hacerlo, se muestra un formulario prellenado con los detalles actuales de la actividad. Este formulario permite al usuario realizar cambios en varios campos, incluyendo el nombre de la actividad, su clasificación, las fechas de inicio y finalización, y si la actividad es individual o grupal. El usuario tiene la opción de cancelar el formulario y la opción de aceptar el formulario. En caso de dar clic en aceptar, la información de la actividad se almacena en la base de datos del sistema. Una vez que el usuario ha realizado los cambios deseados, estos se guardan y actualizan en la base de datos del sistema. Los cambios se reflejan de inmediato, asegurando que el usuario siempre tenga acceso a la información más actualizada.	

Tabla 22. Tarea de Ingeniería: Eliminar actividad

Tarea de Ingeniería	
Nombre de la tarea: Eliminar actividad	
HU: Registrar la actividad	Puntos estimados: 0.2
Fecha de inicio: 27/9/23	Fecha de fin: 27/9/23
Descripción: El usuario tiene la capacidad de eliminar completamente una actividad del sistema. Al dar clic en la opción eliminar, se muestra una interfaz donde el usuario debe de confirmar su acción o cancelarla. En caso de confirmación, la actividad seleccionada se borra tanto de la base de datos del sistema como de la lista de actividades disponibles para los estudiantes. Una vez eliminada, la actividad no volverá a aparecer en el sistema.	

Tabla 23. Tarea de Ingeniería: Registrar participación

Tarea de Ingeniería
Nombre de la tarea: Registrar participación

HU: Registrar participación y resultado de una actividad	Puntos estimados: 0.40
Fecha de inicio: 28/9/23	Fecha de fin: 29/9/23
Descripción: Creación de un registro de participación de un estudiante en una actividad específica. El usuario debe proporcionar los datos necesarios, como la identificación del estudiante y la identificación de la actividad. Una vez que se ingresan estos datos, el registro de participación se guarda en la base de datos.	

Tabla 24. Tarea de Ingeniería: Registrar resultados

Tarea de Ingeniería	
Nombre de la tarea: Registrar resultado	
HU: Registrar participación y resultado de una actividad	Puntos estimados: 0.40
Fecha de inicio: 2/10/23	Fecha de fin: 3/10/23
Descripción: Creación de un registro de los resultados de un estudiante en una actividad específica. El usuario debe proporcionar los datos necesarios, como nombre del premio y fecha. Una vez que se ingresan estos datos, se actualiza el registro de participación en la base de datos.	

Tabla 25. Tarea de Ingeniería: Modificar registro de participación o resultados

Tarea de Ingeniería	
Nombre de la tarea: Modificar registro de participación o resultados	
HU: Registrar participación y resultado de una actividad	Puntos estimados: 0.20
Fecha de inicio: 4/10/23	Fecha de fin: 4/10/23
Descripción: Permite al usuario hacer cambios en el registro de participación o resultados de un estudiante. Pueden actualizar los detalles de la participación o los resultados obtenidos. Una vez que se realizan los cambios, se actualiza la información en la base de datos.	

Tabla 26. Tarea de Ingeniería: Eliminar registro de participación o resultados

Tarea de Ingeniería

Nombre de la tarea: Eliminar registro de participación o resultados	
HU: Registrar participación y resultado de una actividad	Puntos estimados: 0.20
Fecha de inicio: 6/10/23	Fecha de fin: 6/10/23
Descripción: Permite eliminar el registro de participación o resultados de una actividad de un estudiante. Una vez que se selecciona el registro y se confirma la acción, se elimina permanentemente de la base de datos.	

Tabla 27. Tarea de Ingeniería: Insertar sanción

Tarea de Ingeniería	
Nombre de la tarea: Insertar sanción	
HU: Registrar sanción a un estudiante determinado.	Puntos estimados: 0.60
Fecha de inicio: 9/10/23	Fecha de fin: 11/10/23
Descripción: Un formulario se muestra a los usuarios, solicitando detalles específicos para registrar una sanción. Los campos a completar incluyen nombre, clasificación, fecha y número de resolución. El formulario tiene la opción de cancelar y la opción de aceptar una vez llenado todos los campos. En caso del usuario dar clic en aceptar, la información de la sanción se almacena en la base de datos del sistema. Esta sanción registrada se vuelve accesible para los usuarios.	

Tabla 28. Tarea de Ingeniería: Modificar sanción

Tarea de Ingeniería	
Nombre de la tarea: Modificar sanción	
HU: Registrar sanción a un estudiante determinado.	Puntos estimados: 0.40
Fecha de inicio: 12/10/23	Fecha de fin: 13/10/23
Descripción: El usuario tiene la opción de seleccionar una sanción específica del listado de sanciones y modificarla. Al hacerlo, se despliega un formulario prellenado con los detalles	

actuales de la sanción. Este formulario permite al usuario realizar cambios en varios campos, incluyendo nombre, clasificación, fecha y número de resolución. El formulario tiene la opción de cancelar y la opción de aceptar una vez llenado todos los campos. En caso del usuario dar clic en aceptar, la información de la sanción se almacena en la base de datos del sistema. Los cambios se reflejan de inmediato, asegurando que el usuario siempre tenga acceso a la información más actualizada.

Tabla 29. Tarea de Ingeniería: Listar sanciones

Tarea de Ingeniería	
Nombre de la tarea: Listar sanciones	
HU: Registrar sanción a un estudiante determinado.	Puntos estimados: 0.20
Fecha de inicio: 16/10/23	Fecha de fin: 16/10/23
Descripción: Mediante un diseño intuitivo y fácil de usar son mostradas al usuario las sanciones de manera organizada. Son mostrados detalles como el nombre, fecha y número de resolución. En esta interfaz de usuario además aparecen las opciones insertar actividad, modificar actividad y eliminar actividad.	

Tabla 30. Tarea de Ingeniería: Eliminar sanción

Tarea de Ingeniería	
Nombre de la tarea: Eliminar sanción	
HU: Registrar sanción a un estudiante determinado.	Puntos estimados: 0.20
Fecha de inicio: 17/10/23	Fecha de fin: 17/10/23
Descripción: El usuario tiene la capacidad de eliminar completamente una sanción del sistema. Al dar clic en la opción eliminar, se muestra una interfaz donde el usuario debe de confirmar su acción o cancelar la misma. En caso de confirmación la sanción seleccionada se borra tanto de la base de datos del sistema, como de la lista de sanciones de los estudiantes y del estudiante. Una vez eliminada, la sanción no volverá a aparecer en el sistema.	

Tabla 31. Tarea de Ingeniería: Insertar premio o distinción

Tarea de Ingeniería	
Nombre de la tarea: Insertar premio y distinción	
HU: Registrar premios y distinción a un estudiante determinado	Puntos estimados: 0.40
Fecha de inicio: 18/10/23	Fecha de fin: 19/10/23
Descripción: Un formulario se muestra a los usuarios, solicitando detalles específicos para registrar un premio o distinción. Los campos a completar incluyen el nombre, la fecha de entrega. El formulario tiene las opciones cancelar y aceptar una vez llenado todos los campos. En caso del usuario dar clic en aceptar, la información del premio o la distinción se almacena en la base de datos del sistema. Este premio o distinción registrada se vuelve accesible para los usuarios.	

Tabla 32. Tarea de Ingeniería: Listar premio o distinción

Tarea de Ingeniería	
Nombre de la tarea: Listar premio y distinción	
HU: Registrar premios y distinción a un estudiante determinado	Puntos estimados: 0.20
Fecha de inicio: 20/10/23	Fecha de fin: 20/10/23
Descripción: A través de una interfaz de usuario diseñada de manera intuitiva, se muestra al usuario una lista organizada de premio o distinción. Esta lista visualiza detalles de cada premio o distinción, incluyendo el nombre y fecha de entrega. Además, la interfaz proporciona opciones para la gestión de los premios o distinciones. Estas opciones permiten al usuario insertar nuevos premios o distinciones, modificar premios o distinciones existentes y eliminar premios o distinciones según sea necesario. Esta funcionalidad integral facilita la administración eficiente de las actividades por parte del usuario.	

Tabla 33. Tarea de Ingeniería: Modificar premios o distinciones

Tarea de Ingeniería	
Nombre de la tarea: Modificar premios y distinciones	
HU: Registrar premios y distinción a un estudiante determinado	Puntos estimados: 0.40
Fecha de inicio: 23/10/23	Fecha de fin: 24/10/23
Descripción: El usuario tiene la opción de seleccionar de la lista de premios y distinciones una actividad específica que desea modificar. Al hacerlo, se muestra un formulario prellenado con los detalles actuales de los premios o distinciones. Este formulario permite al usuario realizar cambios en varios campos como el nombre, la fecha de entrega. El formulario tiene las opciones cancelar y aceptar una vez llenado todos los campos. En caso del usuario dar clic en aceptar, la información del premio o la distinción se almacena en la base de datos del sistema. Los cambios se reflejan de inmediato, asegurando que el usuario siempre tenga acceso a la información más actualizada.	

Tabla 34. Tarea de Ingeniería: Eliminar premios y distinciones

Tarea de Ingeniería	
Nombre de la tarea: Eliminar premios y distinciones	
HU: Registrar premios y distinción a un estudiante determinado	Puntos estimados: 0.20
Fecha de inicio: 27/10/23	Fecha de fin: 27/10/23
Nombre de la tarea: El usuario tiene la capacidad de eliminar completamente un premio o una distinción del sistema. Al dar clic en la opción eliminar, se muestra una interfaz donde el usuario debe de confirmar su acción o cancelar la misma. En caso de confirmación el premio o la distinción se borra tanto de la base de datos del sistema, como de la lista de premios y distinciones y del estudiante. Una vez eliminado, el premio o distinción no volverá a aparecer en el sistema.	

Tabla 35. Tarea de Ingeniería: Generar caracterización

Tarea de Ingeniería
Nombre de la tarea: Generar caracterización

HU: Generar caracterización del año de un estudiante determinado.	Puntos estimados: 1
Fecha de inicio: 30/10/23	Fecha de fin: 3/11/23
Descripción: Al usuario seleccionar en la lista de estudiantes la caracterización de un estudiante específico. Son mostradas todas las actividades, premios, distinciones y sanciones del mismo durante un año académico.	

Tabla 36. Tarea de Ingeniería: Modificar caracterización

Tarea de Ingeniería	
Nombre de la tarea: Modificar caracterización	
HU: Generar caracterización del año de un estudiante determinado.	Puntos estimados: 1
Fecha de inicio: 30/10/23	Fecha de fin: 3/11/23
Descripción: El usuario puede modificar la caracterización de un estudiante	

Tabla 37. Tarea de Ingeniería: Insertar usuario

Tarea de Ingeniería	
Nombre de la tarea: Insertar usuario	
HU: Gestión de los usuarios del sistema.	Puntos estimados: 0.40
Fecha de inicio: 6/11/23	Fecha de fin: 7/11/23
Descripción: La creación de un nuevo usuario en el sistema. Para ello se debe de crear un nombre de usuario y una contraseña. Una vez que se ingresan estos datos, el usuario se guarda en la base de datos para su posterior referencia.	

Tabla 38. Tarea de Ingeniería: Autenticar usuario

Tarea de Ingeniería	
Nombre de la tarea: Autenticar usuario	
HU: Gestión de los usuarios del sistema.	Puntos estimados: 0.20
Fecha de inicio: 8/11/23	Fecha de fin: 8/11/23
Descripción: Presenta un formulario de autenticación, solicitando al usuario que introduzca sus credenciales, que incluyen el nombre de usuario y la contraseña. Estos datos deben ser	

válidos para permitir la autenticación. Si las credenciales proporcionadas no son correctas, el sistema mostrará un mensaje de error. El acceso a las funcionalidades del sistema estará restringido hasta que el usuario se autentique correctamente.

Tabla 39. Tarea de Ingeniería: Modificar usuario

Tarea de Ingeniería	
Nombre de la tarea: Modificar usuario	
HU: Gestión de los usuarios del sistema.	Puntos estimados: 0.20
Fecha de inicio: 9/11/23	Fecha de fin: 9/11/23
Descripción: Permite al usuario administrador hacer cambios en un usuario existente. Puede cambiar los datos del usuario, como el nombre de usuario, la contraseña. Una vez que se realizan los cambios, se actualiza la información del usuario en la base de datos.	

Tabla 40. Tarea de Ingeniería: Eliminar usuario

Tarea de Ingeniería	
Nombre de la tarea: Eliminar usuario	
HU: Generar caracterización integral del estudiante.	Puntos estimados: 0.20
Fecha de inicio: 10/11/23	Fecha de fin: 10/11/23
Descripción: Al usuario administrador eliminar un usuario existente del sistema. Una vez que se selecciona el usuario y se confirma la acción, el usuario se elimina permanentemente de la base de datos.	

Tabla 41. Tarea de Ingeniería: Generar caracterización integral del estudiante

Tarea de Ingeniería	
Nombre de la tarea: Generar caracterización integral del estudiante	
HU: Generar caracterización integral del estudiante.	Puntos estimados: 1
Fecha de inicio: 13/11/23	Fecha de fin: 17/11/23

Descripción: Al usuario seleccionar en la lista de estudiantes la caracterización de un estudiante específico. A continuación, se muestran todas las actividades, premios, distinciones y sanciones del estudiante durante toda su vida universitaria.

Tabla 42. Tarea de Ingeniería: Modificar caracterización integral

Tarea de Ingeniería	
Nombre de la tarea: Modificar caracterización integral	
HU: Generar caracterización del año de un estudiante determinado.	Puntos estimados: 1
Fecha de inicio: 30/10/23	Fecha de fin: 3/11/23
Descripción: El usuario puede modificar la caracterización de un estudiante para registrar otros elementos no recogidos por el sistema como su decursar académico, misiones, entre otros.	

3.3 Estrategia de pruebas

El objetivo de esta estrategia de pruebas para el sistema de gestión de las actividades educativas de los estudiantes para su desarrollo integral se centra en dos aspectos fundamentales: la verificación de componentes individuales y la validación de la funcionalidad del sistema en su conjunto.

Primero, se realizan **pruebas unitarias** para verificar que cada componente individual del sistema, incluyendo funciones, métodos, clases y módulos, funciona correctamente de manera aislada. Este enfoque permite identificar y solucionar problemas a nivel de componente en las primeras etapas del proceso de desarrollo, mejorando así la eficiencia del desarrollo y la calidad del software.

En segundo lugar, se llevan a cabo **pruebas de aceptación** para asegurar que el sistema cumple con los requisitos y necesidades de los usuarios finales. Estas pruebas se realizarán en un entorno que simula el entorno de producción real, y se centrarán en la funcionalidad y usabilidad del sistema desde la perspectiva del usuario.

3.3.1 Pruebas unitarias

Las pruebas manuales, realizadas al software, buscan confirmar la funcionalidad adecuada de un segmento de código y detectar posibles defectos en su implementación. En este proceso,

se utilizó el método de prueba de camino básico, que implicó la generación de casos de prueba para analizar la complejidad lógica del diseño procedural. Estos casos de prueba se centraron en identificar y establecer un conjunto esencial de rutas de ejecución, proporcionando una visión detallada de la funcionalidad y eficacia del código.

Prueba del Camino básico: se basa en obtener una medida de la complejidad del diseño procedimental de un programa (o de la lógica del programa). Esta medida es la complejidad ciclomática de McCabe, y representa un límite superior para el número de casos de prueba que se deben realizar para asegurar que se ejecuta cada camino del programa.

Los pasos a realizar para aplicar esta técnica son:

- Representar el programa en un grafo de flujo.
- Calcular la complejidad ciclomática.
- Determinar el conjunto básico de caminos independientes.
- Derivar los casos de prueba que fuerzan la ejecución de cada camino.

El grafo de flujo se utiliza para representar flujo de control lógico de un programa. Para ello se utilizan los elementos siguientes:

- Nodos: representan cero, una o varias sentencias en secuencia. Cada nodo comprende como máximo una sentencia de decisión (bifurcación).
- Aristas: líneas que unen dos nodos.
- Regiones: áreas delimitadas por aristas y nodos. Cuando se contabilizan las regiones de un programa debe incluirse el área externa como una región más.
- Nodos predicados: cuando en una condición aparecen uno o más operadores lógicos
- (AND, OR, XOR, ...) se crea un nodo distinto por cada una de las condiciones simples.

```

1  {% for premio in object_list %}
2      <tr>
3          <td><a class="text-decoration-none"
4              href="{% url 'editar_premio' premio.id %}">{{ premio.nombre | escape }}</a>
5          </td>
6          <td> {{ premio.fecha | escape }}</td>
7          <td>
8              <td>
9                  {% if perms.app.change_premio %}
10                     <a id="" class=" mx-1 px-1 text-decoration-none" title="Editar"
11                         href="{% url 'editar_premio' premio.id %}">
12                         
15                     </a>
16                     {% endif %}
17                     {% if perms.app.delete_premio %}
18                         <a id="" class=" mx-1 px-1 text-decoration-none"
19                             title="Eliminar"
20                             href="{% url 'eliminar_premio' premio.id %}">
21                             
24                             </a>
25                         {% endif %}
26                     </td>
27                 {% endif %}
28             </td>
29         </tr>
30     {% endfor %}

```

Diagrama de flujo del código:

- Nodo 1: Inicio del bucle `{% for premio in object_list %}`.
- Nodo 2: Inicio de la etiqueta `<tr>`.
- Nodo 3: Inicio de la etiqueta `<td>` que contiene el enlace de edición.
- Nodo 4: Inicio de la etiqueta `<td>` que contiene la fecha.
- Nodo 5: Inicio de la etiqueta `<td>` que contiene los botones de acción.

Figura 4: Código del método de prueba generar sanción

El grafo de flujo del programa es una representación gráfica del código donde los nodos simbolizan las instrucciones y las aristas indican las rutas de ejecución posibles. Para determinar la complejidad ciclomática, se debe crear primero el grafo de flujo del programa y contar el número de aristas, nodos y componentes conectados. Posteriormente, se utiliza la fórmula de McCabe para calcular el valor de M.

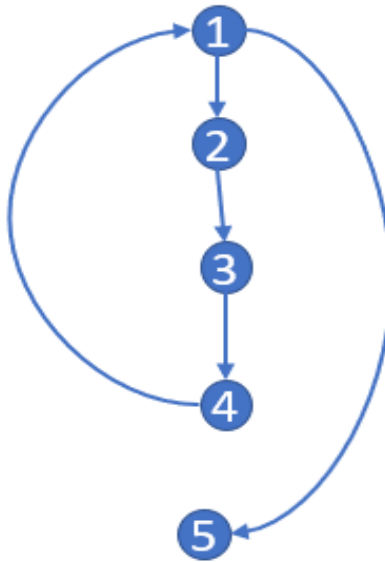


Figura 5: Grafo de flujo

Cálculo de la complejidad ciclomática:

$$V(G) = E - N + 2$$

$$V(G) = 5 - 5 + 2 = 2$$

Rutas independientes:

R1: 1, 2, 3, 4

RI2: 1, 5

Según los resultados, la funcionalidad posee una complejidad ciclomática de 2, lo que indica que existen hasta 2 caminos lógicos de ejecución. Esto sugiere que la funcionalidad representa un riesgo bajo para el sistema.

3.3.2 Pruebas de aceptación

Las pruebas de aceptación, que se realizan principalmente utilizando técnicas de caja negra, son un nivel esencial de pruebas en el proceso de desarrollo de software. Estas pruebas son definidas por el cliente para cada historia de usuario. El cliente tiene la responsabilidad de realizar estas pruebas, asegurándose de que el sistema responda adecuadamente a diversas entradas de datos y cumpla con las funcionalidades deseadas.

Para llevar a cabo estas pruebas de manera efectiva, se utiliza la técnica de Partición Equivalente. Un caso de prueba ideal revela de inmediato una clase de errores que, de lo

contrario, requerirían muchos casos para detectar el error genérico. La partición equivalente se enfoca en la definición de casos de prueba que descubren clases de errores, minimizando así el número total de casos de prueba que deben desarrollarse.

Una clase de equivalencia representa un conjunto de estados válidos o inválidos para las condiciones de entrada. Normalmente, una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica.

Las pruebas de aceptación del sistema fueron ejecutadas por los clientes, quienes, al ser profesionales en ingeniería informática, poseen un profundo entendimiento y experiencia en la realización de este tipo de pruebas. Esta colaboración directa con los usuarios finales, que son expertos en la materia, no solo asegura que el sistema cumple con sus necesidades específicas, sino que también permite una evaluación más precisa y relevante de su funcionalidad y rendimiento.

A continuación, se muestra un ejemplo de pruebas de aceptación realizado por los clientes:

Tabla 43. Caso de prueba de la tarea de ingeniería: Insertar la actividad

Escenario	Descripción	Nombre	Fecha de inicio	Fecha de fin	Clasificación	Tipo	Respuesta del sistema	Flujo Central
EC 1.1 Insertar una actividad	Se inserta una actividad en la lista de actividades.	V	V	V	V	V	El sistema inserta la nueva actividad	1- Seleccionar el botón "Insertar actividad".
		Bastión Estudiantil Universitario	21/11/2023	21/11/2023	Político-Ideológico	Grupal		
EC 1.2 Insertar una actividad	Datos obligatorios vacíos.	I	V	V	V	V	El sistema muestra el mensaje: "Rellene este campo".	2- Introducir datos en el formulario. 3- Presionar el botón
			21/11/2023	21/11/2023	Político-Ideológico	Grupal		
		V	I	V	V	V		
		Bastión Estudiantil		21/11/2023	Político-Ideológico	Grupal		

		Universitario						"Aceptar".
		V	V	I	V	V		
		Bastión Estudiantil Universitario	21/11/2023		Político-Ideológico	Grupal		
		V	V	V	I	V		
		Bastión Estudiantil Universitario	21/11/2023	21/11/2023		Grupal		
		V	V	V	V	I		
		Bastión Estudiantil Universitario	21/11/2023	21/11/2023	Político-Ideológico			
EC 1.3 Datos incorrectos.	Comprobar que no haya datos incorrectos	I	V	V	V	V	El sistema muestra el mensaje: "El campo contiene caracteres no permitidos."	
		Bastión Estudiantil Universitario	21/11/2023	21/11/2023	Político-Ideológico	Grupal		
		V	I	V	V		El sistema muestra el mensaje: "El tiempo de inicio no puede ser mayor que el tiempo de fin."	
		Bastión Estudiantil Universitario	21 de 11 de 2023	21/11/2023	Político-Ideológico	Grupal		
		V	V	I	V	V		

		Bastión Estudiantil Universitario	21/11/2023	21 de 11 de 2023	Político-Ideológico	Grupal	El sistema muestra el mensaje: "El formato de la fecha es dd/mm/aaaa."	
		V	V	V	I	V	El sistema muestra el mensaje: "Debe seleccionar al menos un elemento."	
		Bastión Estudiantil Universitario	21/11/2023	21/11/2023		Grupal		
		V	V	V	V	I		
		Bastión Estudiantil Universitario	21/11/2023	21/11/2023	Político-Ideológico		El sistema muestra el mensaje: "Debe seleccionar al menos un elemento."	
EC 1.4 Acción cancelar	Permite cancelar la opción de insertar una actividad	V	V	V	V	V	El sistema cancela la acción y cierra la ventana.	1- Seleccionar el botón "Insertar actividad". 2- Presionar el botón "Cancelar".
		N/A	N/A	N/A	N/A	N/A		

3.4 Resultados de las pruebas

Durante la fase de pruebas, se llevaron a cabo tres iteraciones, con el objetivo principal de determinar el grado de cumplimiento de la aplicación con las funcionalidades especificadas. La estrategia permitió evaluar de manera efectiva la eficacia de la aplicación en términos de satisfacer las expectativas funcionales establecidas.

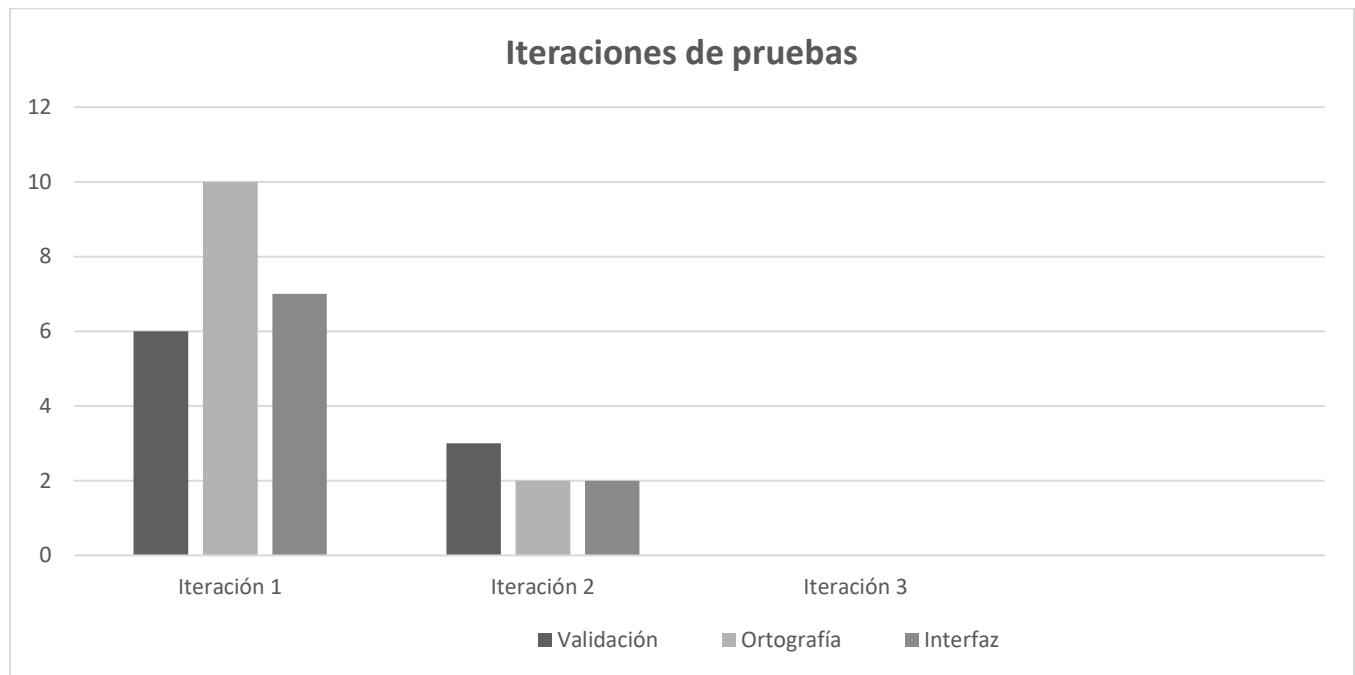


Figura 6. Resultados de las pruebas

En una primera iteración se identificaron un total de 23 No Conformidades, 6 a errores de validación, 10 asociadas a errores ortográficos y 7 a errores de interfaz. Luego de corregidas estas deficiencias se procedió a la realización de una segunda iteración de pruebas donde se encontraron 5 No Conformidades, 3 a errores de validación, 2 asociadas a errores de interfaz y 2 a errores ortográficos. Todas se corrigieron y se realizó una tercera iteración de pruebas donde no fueron encontradas No Conformidades, por lo que se determina no realizar nuevas iteraciones.

Conclusiones del capítulo

- Se conformaron 24 tareas de ingeniería
- Se definió la estrategia de prueba a seguir que permitió la verificación de las funcionalidades del sistema.

- La aplicación de pruebas unitarias y pruebas de aceptación posibilitó verificar el correcto funcionamiento del sistema.
- El proceso de pruebas permitió identificar y corregir las No Conformidades identificadas en el sistema.
-

Conclusiones generales

Esta investigación se centró en el desarrollo de un sistema de gestión para las actividades educativas de los estudiantes enfocado en su desarrollo integral. Para lograr este objetivo, se establecieron y cumplieron los objetivos específicos planteados en la introducción. Los resultados obtenidos fueron satisfactorios, lo que nos lleva a las siguientes conclusiones:

- La formulación del marco teórico-metodológico que apoya el desarrollo y aplicación de un sistema de gestión para las actividades educativas de los estudiantes, orientado hacia su desarrollo integral, ha subrayado la relevancia de la digitalización de los procesos que se ejecutan en la Universidad de las Ciencias Informáticas (UCI).
- Las siete historias de usuarios identificadas permitieron el desarrollo del sistema de gestión de actividades educativas de los estudiantes para su desarrollo integral.
- Las pruebas y correcciones realizadas en cada iteración permitieron la realización de un software de calidad.

Referencias Bibliográficas

Alarcón Ortiz, R. A., Guzmán Mirás, Y., & García González, M. (2019). Formación integral en la educación superior: una visión cubana.

Asana. (2022). ¿Qué es la programación extrema (XP)? <https://asana.com/es/resources/extreme-programming-xp>

Carlos Álvarez de Zayas (1998). Metodología de la Investigación. / Carlos A. de Zayas, Virginia Sierra Lombardía. -- La Habana.: MES:1998.

Fuentes, H., Peña, R., & Milán, M. R. (2010). La evaluación del Proceso Docente Educativo como proceso participativo.

García M., S. (2015). Django, la guía definitiva. Django Software Corporation.

Guzmán, Y. & García, M. (2017). La competencia educativa del coordinador de año en el contexto de la nueva universidad cubana. Revista Cubana de Educación Superior.

HTML (s. f.). Introducción al HTML. <https://html.spec.whatwg.org/multipage/>.

Menzinsky, A., López, G., Palacio, J., Sobrino, M. Á., Álvarez, R., & Rivas, V. (2022). Historias de Usuario: Ingeniería de Requisitos Ágil.

Ministerio de Educación Superior. (2014). Perfeccionamiento del sistema de gestión del proceso de formación integral de estudiantes universitarios en el eslabón de base. Félix Varela.

Ministerio de Educación Superior. (2018). Reglamento de Trabajo Docente y Metodológico de la Educación Superior. Resolución No. 2/2018. MES.

Ministerio de Educación Superior. (2022). Reglamento Organizativo del Proceso Docente y de dirección del Trabajo Docente y Metodológico para las carreras universitarias. Resolución 47/2022. MES.

MDN Web Docs. (2022). HTML5 - Glosario de MDN Web Docs: Definiciones de términos relacionados con la Web <https://developer.mozilla.org/es/docs/Glossary/HTML5>

Mozilla. (s. f.). Introducción a Django - Aprende desarrollo web. MDN Web Docs. https://developer.mozilla.org/es/docs/Learn/Server-side/Django/Introduction#%C2%BFqu%C3%A9_es_django

JetBrains. (2023). Pycharm, “Guía de inicio rápido”.
<https://www.jetbrains.com/help/pycharm/quick-start-guide.html>

Pressman, R. S. (2010). Ingeniería de Software: Un enfoque práctico (7ma edición).

Ponjuan, G. 2011. La gestión de información y sus modelos representativos.

Python. (2023). El tutorial de Python. <https://docs.python.org/3/tutorial/index.html>

Ramos Acevedo, I. N., Massip Acosta, A., Alfonso Nazco, M., & Ronda Rodríguez, N. (2020). La evaluación integral de estudiantes universitarios, expresión de calidad en su formación: apuntes y reflexiones.

Ramos, I., Massip, A., & Alfonso, M. (2017). La evaluación integral del estudiante como vía para estimular el protagonismo en su formación. Universidad y Sociedad.

Sommerville, I. (2011). Ingeniería de software (9na edición).

Vidal Ledo, María, Durán García, Francisco, & Pujal Victoria, Nayra. (2008). Gestión educativa. Educación Médica Superior. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21412008000200012&lng=es&tlng=.

Visual Paradigm. (s.f.) ¿Qué hace Visual Paradigm? <https://www.visual-paradigm.com/aboutus/faq>.