Universidad de las Ciencias Informáticas
Facultad 4

Módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria (SIGEU)

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

Autor: Airely Sanchez Miralles

Tutores: MSc. Julio César Espronceda Pérez

Ing. Alvaro Alejandro Acosta Ruiz

La Habana, 2023

Dime y lo olvido, enséñame y lo recuerdo, involúcrame y lo aprendo.

Benjamin Franklin

Dedicatoria

A mis abuelos, que me enseñaron el valor del trabajo, la perseverancia y la humildad. Aunque tres de ellos ya no estén físicamente conmigo, siempre los llevo en mi corazón y en mi mente, significando la luz que nunca me ha faltado ni me faltará para siempre aclarar mis caminos y mis objetivos como el hombre de bien que siempre quisieron que fuera. Gracias por su amor y su ejemplo.

A mi familia, que me ha apoyado incondicionalmente en este camino. A mi hermano, a mis tíos, primos y amigos, que me han brindado su confianza, su ánimo y su alegría. Sin ustedes, nada de esto hubiera sido posible.

Y especialmente, a mis padres, que son mi inspiración y mi orgullo. Gracias por darme la vida, la educación y la oportunidad de cumplir mis sueños. Gracias por creer en mí, por acompañarme y por sacrificarse por mi bienestar. Son los mejores padres del mundo y les dedico este triunfo, y todos los que siempre pueda cumplir.

Esta tesis es el resultado de muchos años de esfuerzo, dedicación y pasión por esta carrera. Espero que sea mi granito de arena al conocimiento y al desarrollo de futuros proyectos que tributen a la informatización de los procesos en la Universidad. Agradezco a todos los que me han guiado, ayudado y motivado en este proceso. Estoy muy feliz y orgulloso de haber llegado hasta aquí.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas las personas que han hecho posible la realización de esta tesis.

En primer lugar, a mis tutores, Álvaro y Julio, por su excelente orientación, paciencia y apoyo. Gracias por compartir sus conocimientos, experiencias y consejos conmigo. Han sido unos maestros excepcionales y unos grandes profesionales.

A mis amigos, Arthur, José Adrián y mi hermano de crianza Chrystian, por su amistad, su compañerismo y su solidaridad. Gracias por estar siempre ahí, por escucharme, por animarme y por formar parte de los momentos más divertidos y más importantes a lo largo de esta travesía. Han sido unos cómplices inolvidables y unos verdaderos hermanos.

A los compañeros de trabajo de mi papá, por su colaboración, su generosidad y disposición. Gracias por facilitarme el acercamiento con mi familia a través de la distancia. Han sido unos aliados valiosos y unos referentes admirables. Por ustedes también hoy estoy realizando mi culminación de estudios.

Y por último, a mis padres, por todo el sacrificio que sin dudar un momento tuvieron conmigo. Gracias por darme siempre el ánimo que necesité para estudiar y superarme, por apoyarme en todo momento, por motivarme a seguir adelante y por estar orgullosos de mí. Siempre serán mi ejemplo a seguir y les debo todo lo que soy.

Declaración de autoría

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los 29 días del mes de noviembre del año 2023.

Airely Sanchez Miralles
Autor

MSc. Julio César Espronceda Pérez
Tutor

Ing. Alvaro Alejandro Acosta Ruiz
Tutor

Resumen

Las universidades, son instituciones culturales que permiten preparar al hombre para su labor en la sociedad, tiene que ser transformadora y dispuesta al cambio, a la creación de soluciones y corrientes ideológicas que definan a una sociedad. La extensión universitaria constituye uno de los procesos principales de la educación superior en Cuba. Cada universidad cuenta dentro de su estructura organizativa con la Dirección de Extensión Universitaria, que se encarga de dirigir el proceso extensionista. La Universidad de las Ciencias Informáticas como institución perteneciente al Ministerio de Educación Superior de Cuba no está exento a ello, por lo tanto, además de organizar y dirigir el proceso extensionista, gestiona la información asociada a las actividades deportivas, el cual se considera como uno de los subprocesos denominado Movimiento Deportivo. Dentro del Sistema Gerencial de Extensión Universitaria, sistema informático que actualmente se encuentra en desarrollo se cuenta con la gestión de otros subprocesos de la extensión universitaria, sin embargo, esta gestión posee numerosas dificultades, que imposibilitan su correcto funcionamiento. La presente investigación tiene como objetivo desarrollar un módulo informático que contribuya a la gestión de la información asociada al movimiento deportivo en la Universidad de las Ciencias Informáticas perteneciente al sistema antes mencionado. Para el desarrollo de la propuesta se utilizó como metodología Proceso Unificado Ágil (AUP, por sus siglas en inglés) en su versión Universidad de las Ciencias Informáticas (UCI), escenario 4, de manera que, a partir de la investigació se definieran los requisitos funcionales y se establecieran acorde a dicha metodología así como el diseño a partir de las herramientas Balsamiq y Visual Paradimg, una vez definidas estos actefactos se establece la lócoca del negocio definiendo las principales funcionalidades y para su implementación se utiliza el framework de desarrollo Django y como lenguaje de desarrollo se utilizó Python. Para evaluar la calidad de la solución se realizaron pruebas de software donde se pudo corroborar el cumplimiento de los requisitos acordados con el cliente. Por último, para medir el nivel de satisfacción de los usuarios se aplicó la técnica de Iadov donde se obtuvieron resultados satisfactorios.

Palabras clave: actividades extensionistas, cultura, deporte, extensión, gestión.

Índice general

Introducción **1**

1 Marco teórico referencial sobre módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria **5**

1.1 Conceptos asociados a la investigación 5

1.1.1 Extensión universitaria 5

1.1.2 Dirección de Extensión Universitaria 6

1.1.3 Movimiento Deportivo 6

1.1.4 Gestión de información 7

1.2 Estudio de sistemas homólogos 8

1.2.1 Ámbito internacional 8

1.2.2 Ámbito nacional 8

1.2.3 Consideraciones del estudio de sistemas homólogos 10

1.3 Metodologías, herramientas, y tecnologías 10

1.3.1 Modelado de software 10

1.3.2 Herramientas y tecnologías 11

1.4 Conclusiones del capítulo 14

2 Análisis y diseño del módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria **15**

2.1 Descripción de la propuesta de solución 15

2.2 Especificación de requisitos del sistema 16

2.2.1 Requisitos funcionales 16

2.2.2 Requisitos no funcionales 17

2.3 Descripción de las historias de usuario 18

2.4 Estilo arquitectónico 20

2.4.1 Patrón arquitectónico 21

2.5 Diagrama de clases de diseño 22

2.6 Patrones de diseño 23

2.7 Modelo de datos 25

2.8 Consideraciones parciales 26

3 Implementación y pruebas del módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria 28

3.1 Estándares de codificación 28
3.2 Diagrama de despliegue 31
3.3 Pruebas de software 32
3.3.1 Estrategia de pruebas 32
3.3.2 Pruebas unitarias 33
3.3.3 Pruebas funcionales 36
3.3.4 Pruebas de Integración 39
3.3.5 Prueba de seguridad 40
3.3.6 Prueba de rendimiento 42
3.3.7 Pruebas de aceptación 43
3.4 Satisfacción de los usuarios 44
3.5 Consideraciones parciales 45

Conclusiones 47

Recomendaciones 48

Acrónimos 49

Referencias bibliográficas 51

Apéndices 54

.1 Modelo de encuesta aplicada a especialistas 55
.1.1 Resultados de la encuesta aplicada a especialistas 55
.2 Guía de observación para el análisis de los sistemas similares 56
.3 Historias de usuario 56

Índice de figuras

2.1 Representación de la arquitectura. (Fuente: Elaboración propia) 21
2.2 Diagrama de clases de gestión de Delegaciones Deportivas 22
2.3 Modelo de datos 25

3.1 Representación de la arquitectura. (Fuente: Elaboración propia) 31
3.2 Prueba de seguridad. (Fuente: Elaboración propia) 36
3.3 Representación del resultado de las pruebas. (Fuente: Elaboración propia) 37
3.4 Representación del resultado de las pruebas. (Fuente: Elaboración propia) 38
3.5 Representación de la integración del directorio del Módulo en cuestión al proyecto. (Fuente: Elaboración propia) 39
3.6 Integración Descendente. (Fuente: Elaboración propia) 40
3.7 Prueba de seguridad. (Fuente: Elaboración propia) 41
3.8 Prueba de seguridad. (Fuente: Elaboración propia) 41
3.9 Prueba de rendimiento. (Fuente: Elaboración propia) 42
3.10 Prueba de rendimiento. (Fuente: Elaboración propia) 42

Índice de tablas

2.1 Descripción de los Requisitos Funcionales. (Fuente: Elaboración propia). 16
2.2 Historia de usuario # 1 18
2.3 Historia de usuario # 2 19

3.1 Estándares de codificación a utilizar en la implementación del sistema. (Fuente: Elaboración propia). 28
3.2 Estrategia de pruebas. (Fuente: Elaboración propia). 32
3.3 Cálculo de la complejidad ciclomática del método post de la clase Agregar_Evento 34
3.4 Caminos del grafo de flujo (Fuente: Elaboración propia). 35
3.5 Caso de Prueba para el camino básico 1 (Fuente: Elaboración propia). 35
3.6 Caso de prueba “Añadir Integrante” (Fuente: Elaboración Propia). 37
3.7 Variables de caso de prueba “Añadir Integrantes a una Delegación Deportiva”(Fuente: Elaboración Propia). 38
3.8 Cuadro Lógico de Iadov (Fuente: Elaboración propia). 44
9 Encuesta para aplicar Iadov 55
10 Guía de observación para estudio de soluciones (Fuente: Elaboración propia). 56
11 Historia de usuario # 3 56
12 Historia de usuario # 4 57
13 Historia de usuario # 5 58
14 Historia de usuario # 6 58
15 Historia de usuario # 7 59
16 Historia de usuario # 8 59
17 Historia de usuario # 9 60
18 Historia de usuario # 10 61
19 Historia de usuario # 11 61
20 Historia de usuario # 12 62
21 Historia de usuario # 13 63
22 Historia de usuario # 14 63
23 Historia de usuario # 15 64
24 Historia de usuario # 16 65

25 Historia de usuario # 17 65
26 Historia de usuario # 18 66
27 Historia de usuario # 19 67
28 Historia de usuario # 20 68
29 Historia de usuario # 21 69
30 Historia de usuario # 22 69
31 Historia de usuario # 23 70
32 Historia de usuario # 24 70
33 Historia de usuario # 25 71
34 Historia de usuario # 26 71
35 Historia de usuario # 27 72

Introducción

Como resultado de los avances de la sociedad y la constante búsqueda del perfeccionamiento de los procesos que impulsan el desarrollo, el hombre se ha dado a la tarea de buscar desde diferentes posturas, la manera de gestionar más eficiencia, productividad y calidad para garantizar ese empeño. La actual competitividad, presente en todos los escenarios tecnológicos, ha sido un elemento decisivo para que las organizaciones adecuen sus procesos y los adapten a las necesidades de los clientes y demás partes interesadas.

Las universidades no escapan a lo antes expuesto, son reconocidas cada vez más como un instrumento de desarrollo de ciudades, regiones, países, y están consideradas como un factor clave para incrementar la competitividad entre las organizaciones profesionales y la calidad de vida. El desafío para las instituciones de Educación Superior es el de enfrentar un mundo, en el cual los sistemas tecnológicos-productivos están en permanente transformación. Los cambios en las comunicaciones han modificado la forma de percibir el tiempo y las distancias, a la vez que abren nuevas perspectivas para la docencia y la investigación. La influencia del medio externo las compromete, cada vez más, a dirigir sus esfuerzos de gestión hacia una nueva forma de actuación en el proceso de formación de sus profesionales, a cambiar la estructura tradicional por nuevos diseños organizacionales, especialmente para enfrentar la incertidumbre ante las exigencias y las consecuencias de la globalización. En este sentido, se hace imprescindible la necesidad de buscar herramientas organizacionales y funcionales dentro de las Tecnologías de la Información y las Comunicaciones (TIC) que impulsen el desarrollo de la gestión para la universidad, y así obtener la calidad, como un objetivo supremo en la comunidad universitaria (Castellón y Díaz, 2020).

La introducción y el uso de las TIC ha significado a escala mundial un salto vertiginoso en el desarrollo científico técnico; desde su llegada a los escenarios nacionales se han convertido en un elemento indispensable para establecer las líneas de desarrollo de la sociedad cubana, buscando dar solución a los problemas del hombre, y han transformado nuestra manera de trabajar liberándonos de las cargas más pesadas, optimizando los recursos y haciéndonos más productivos (Espronceda Pérez; Pérez Fuentes y Remedio Frometa, 2021). La Informatización de la Sociedad y la Transformación Digital se define en Cuba como el proceso de utilización ordenada y masiva de las nuevas TIC para satisfacer las necesidades de información y conocimiento de la sociedad. Dicha estrategia impulsada por el gobierno posibilita potenciar los logros que en estas áreas ha logrado nuestra Revolución y facilita a nuestra sociedad acercarse más hacia el objetivo de un desarrollo justo, equitativo, sostenible y alcanzable (González Reyes y Febles Estrada, 2021).

La universidad es una institución cultural con una marcada responsabilidad con la sociedad que se concreta en actuar por el mejoramiento económico, político, social y cultural como sistema de vida del pueblo. No

está conformada únicamente con el desarrollo de sus procesos de docencia e investigación, sino que también requiere desarrollar el proceso de extensión para dar cumplimiento a su encomienda social, promoviendo la elevación del nivel cultural a partir de la participación del hombre como agente activo de su propio desarrollo (MES, 2004).

El Ministerio de Educación Superior (MES) diseñó el Programa Nacional de Extensión Universitaria para regir las pautas del proceso de extensión en las universidades del país. Donde a través de la labor educativa, político-ideológica y cultural se forman jóvenes comprometidos con la sociedad. El programa está concebido con una máxima flexibilidad en su diseño y aplicación, de forma tal, que se adapte a la dinámica y las condiciones del entorno.

Dentro de los centros de estudios pertenecientes al MES se encuentra la UCI con la misión de formar profesionales altamente calificados, con una alta formación cultural, comprometidos con la patria y la informatización de la sociedad cubana. Para ello, cuenta con varias vicerrectorías; entre las que se encuentra la Vicerrectoría de Extensión Universitaria y Residencia. Desde el ámbito de la extensión universitaria se plantean diferentes manifestaciones como el deporte, la cultura, la comunicación, la recreación sana y la formación de una cultura general e integral de la comunidad en interacción con la sociedad (Campmany Cejas, 2022). Las cuales son organizadas desde la Vicerrectoría de Extensión Universitaria en coordinación con las facultades docentes y las organizaciones estudiantiles y sindicales, contando con la colaboración de profesores y entrenadores de educación física, atletismo y varias disciplinas deportivas quienes imparten clases y entrenamientos para sus aspirantes.

Los juegos deportivos de la Federación Estudiantil Universitaria (FEU) son la esencia del movimiento deportivo dentro de las universidades, dicho movimiento comenzó cuando se realizó el primer acto nacional por el día de la Cultura Física y el Deporte el 16 de noviembre de 1985 en el estadio vuelta bajero Capitán San Luis con la presencia del entonces vicepresidente del consejo de estado y de ministros José Ramón Fernández Álvarez y una amplia representación de los diferentes sectores educacionales, así como trabajadores. En él se premian no solo el esfuerzo y la dedicación de los deportistas aficionados, sino también profesores, directivos y entrenadores, sin los cuales no fuera posible exhibir el prestigio que le ha merecido el respaldo incondicional del Instituto Nacional de Deporte, Educación Física y Recreación y su red de consejos, instituciones y organizaciones deportivas (MES, 2022).

En la UCI, como parte de la infraestructura para garantizar el desarrollo del movimiento meportivo se encuentra el Gimnasio Polivalente, dirigido por la Dirección de Extensión. Este dispone de salas especializadas para la gimnasia, levantamiento de pesas, judo, lucha y boxeo. Además, cuenta con canchas de Voleibol, Basquetbol, Futsal, Balonmano, así como un terreno de Beisbol y futbol 11, brindando así la calidad requerida para el desarrollo de los juegos deportivos.

En la realización de los juegos deportivos Mella participan aficionados al deporte en diferentes disciplinas, de los eventos se registran sus datos de manera manual y en formato duro, en su generalidad sin un estandar de registro. En sus enfrentamientos se recogen datos tales como los resultados obtenidos, el nombre de la competición y evidencias que son almacenadas y pueden ser utilizadas en posteriores realizaciones de juegos

deportivos. Además, son necesarias en caso de que dichos deportistas cualifiquen para participar en juegos deportivos de carácter provincial o nacional. También es fundamental para divulgar la cantidad de ganadores y participantes para publicarlos en la comunidad universitaria, permitiendo conocer datos como la facultad con mayores disciplinas premiadas. Esta información es llevada de forma manual; lo que ocasiona que se realice varias veces a causa de pérdidas de la información. Requiere de la inversión de tiempo, recursos humanos y materiales. Se ve agudizada al guardar la misma en archivos digitales no estandarizados, en una computadora provocando que en ocasiones no esté disponible, además se hace sumamente engorroso generar cualquier tipo de reporte, analizar y obtener algún tipo de estadística en un año, facultad o disciplina. Actualmente para llevar a cabo la gestión de la información en la Extensión Universitaria se cuenta con un sistema informático destinado a dicho proceso, el cual trabaja encaminado a los diferentes procesos de la Extensión Universitaria, definidos por módulos del sistema, tales como, el Movimiento de Artistas Aficionados, las Instituciones Culturales, las Cátedras Honoríficas, entre otros, pero dicho sistema no cuenta con una herramienta o prestación que permita gestionar el Movimiento Deportivo, dentro de dicho movimiento se realizan varias actividades de las que se necesita gestionar y conservar la información.
Por lo que se llega a el siguiente **problema de investigación**: ¿Cómo contribuir con la gestión de la información del movimiento deportivo universitario en el Sistema Gerencial de Extensión Universitaria?
A partir del problema de investigación formulado, se define como **objeto de estudio**: la gestión de la información, y dentro de este se precisa como **campo de acción**: gestión de la información del movimiento deportivo universitario.
Para darle solución al problema antes planteado, se define como **objetivo general**: Desarrollar un módulo que contribuya con la gestión de la información del movimiento deportivo universitario en el Sistema Gerencial de la Extensión Universitaria. Del mismo se derivan los siguientes **objetivos específicos**:

1. Elaborar del marco teórico de la investigación a partir del estado del arte existente sobre la gestión del Movimiento Deportivo en la UCI.
2. Revisión sistemática y profunda de la bibliografía.
3. Seleccionar de la metodología, herramientas de software a utilizar y lenguajes de programación para el desarrollo de un módulo para la gestión de la información para el movimiento deportivo en la UCI.
4. Diseño y análisis el módulo para la gestión de la información para el movimiento deportivo en la UCI .
5. Implementar el módulo para la gestión de la información asociada al movimiento deportivo en la UCI.
6. Validar el módulo a desarrollar para garantizar la calidad de la gestión del movimiento deportivo en la UCI. Elaborar de pruebas a partir de los resultados obtenidos que aseguren la calidad del módulo a desarrollar.
7. Integrar el módulo movimiento deportivo al Sistema Gerencial de Extensión Universitaria de la UCI.

Para darle solución a los objetivos trazados en la investigación se aplicaron los siguientes **Métodos Científicos**:
Métodos teóricos

- Histórico-Lógico: se emplea para identificar posibles funcionalidades que pueda tener el sistema a partir del análisis de la evolución de los sistemas de gestión para el Movimiento Deportivo.
- Análisis-Síntesis: empleado para el análisis, evaluación y selección de las técnicas a emplear en el desarrollo del sistema. Así como para sintetizar la información que se obtuvo mediante la entrevista con el cliente de manera que pudiera ser usada en el desarrollo del mismo, además, en la identificación de los elementos del marco teórico de la investigación.
- Modelación: para realizar una representación del proceso estudiado que sirva de guía en el desarrollo del sistema, y mediante este, identificar las características y relaciones fundamentales.

Métodos empíricos

- Entrevista: empleado en los encuentros con el cliente para obtener la información necesaria que permita determinar las características, cualidades y requisitos con los que debe contar la propuesta de solución.
- Observación: empleado para obtener el conocimiento necesario del funcionamiento de las soluciones existentes para la gestión de actividades extensionistas de las Instituciones de Educación Superior (IES) cubanas.

El presente trabajo de diploma está estructurado de la siguiente manera: introducción, tres capítulos, conclusiones generales, recomendaciones y las referencias bibliográficas empleadas durante el desarrollo de la investigación, y por último para complementar la investigación se presentan una serie de anexos. A continuación, se muestra la descripción de los capítulos:

- **Capítulo 1.** Marco teórico referencial sobre el sistema para la gestión de la información del Movimiento Deportivo en la UCI: se realiza un análisis del estado del arte sobre los principales aspectos de la investigación. Se fundamenta la selección del proceso de desarrollo de software, así como las tendencias, técnicas, metodología y tecnologías usadas en la propuesta de solución.
- **Capítulo 2.** Características y diseño del sistema para la gestión de la información del Movimiento Deportivo en la UCI: este capítulo aborda los principales aspectos relacionados con los requisitos de la propuesta de solución. Se identifican los requisitos funcionales y no funcionales con los que debe cumplir la solución propuesta, así como estilo arquitectónico, los diagramas de clases, modelo de datos y los patrones de diseño para lograr buenas prácticas en el diseño y posterior implementación del sistema. Igualmente se muestran los principales artefactos de ingeniería de software propuestos por la metodología utilizada.
- **Capítulo 3.** Implementación y pruebas del sistema para la gestión de la información del Movimiento Deportivo en la UCI: en este capítulo se define el estándar de codificación que sirve de guía para la implementación de la solución propuesta, así como la estrategia de pruebas a aplicar para lograr un correcto funcionamiento.

CAPÍTULO 1

Marco teórico referencial sobre módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria

En el presente capítulo se analizan los conceptos que resultan útiles para un mejor entendimiento de los términos asociados al problema planteado y campo de acción, además de un estudio de los sistemas similares y cómo tributan estos a la presente solución. Posteriormente, se fundamentan las tecnologías, herramientas y metodologías utilizadas en el desarrollo del sistema para la gestión de la Extensión Universitaria en la UCI. Se realiza una profunda revisión sistemática de la bibliografía con el fin de lograr una mayor comprensión del alcance de la investigación. En este capítulo se aborda el estado del arte del tema propuesto y los conceptos asociados con los sistemas de gestión de la información.

1.1. Conceptos asociados a la investigación

Para llevar a cabo la investigación es necesario analizar los conceptos asociados al objeto de estudio y campo de acción de la investigación..

1.1.1. Extensión universitaria

La Extensión Universitaria (EU) es el proceso sustantivo de toda Institución de la Educación Superior que se identifica con el rol de promover la cultura, en su más amplia aceptación, en la comunidad intra y extra-universitaria, la cual se articula con la formación y la investigación y junto con estas contribuye al desarrollo sociocultural y de los valores (González Aportela; Batista Mainegra y González Fernández-Larrea, 2020). Es considerada, en el contexto de las universidades, como una de las funciones esenciales que, conjuntamente con la investigación y la docencia, constituyen los pilares básicos sobre los que se construye un modelo de universidad democrática y comprometida socialmente, en su permanente búsqueda hacia la excelencia, pertinencia y equidad social.
Según Kaplún, 2013, la EU como proceso sustantivo de las universidades, es un conjunto de actividades donde colaboran actores intra y extra-universitarios, de forma tal que se aporten sus respectivos saberes, y

aprendan en un proceso interactivo orientado a la expasión de la cultura, llevándola, con prioridad hacia los sectores socialmente más rezagados, sin embargo en Cuba, el MES, mediante el Programa Nacional de Extensión Universitaria (PNEU), se define como una manifestación de la relación dialéctica entre la universidad y la sociedad, pero hace la aclaración, de que esa relación debe ser solamente con el objetivo de promover la cultura y el conocimiento (MES, 2004).

La EU juega un gran papel dentro de las universidades, debido a que su principal función es salvaguardar, desarrollar y promover la cultura. Dentro de la UCI se cuenta con varios procesos como son el Movimiento Deportivo, el Movimiento de Artistas Aficionados, las Cátedras Honoríficas, los Cursos de Capacitación y los Proyectos Comunitarios, los cuales permiten elevar el nivel cultural de la comunidad y al desarrollo de los estudiantes profesores y trabajadores.

El autor de la presente investigación a partir del análisis realizado y tomando como referente la definición de varios autores, entiende como proceso de extensión universitaria el conjunto de actividades intra y extra universitarias relacionadas con los procesos de la educación superior que se realizan con el objetivo de promover cultura y el deporte.

1.1.2. Dirección de Extensión Universitaria

La Dirección de Extensión Universitaria (DEU) es el órgano encargado metodológicamente de coordinar, promover y fomentar el desarrollo de la extensión universitaria para toda la educación superior (ibíd.). Dentro de sus líneas de trabajo cuenta con la creación y desarrollo de proyectos socioculturales comunitarios, además de impulsar la creación y la promoción del arte y el deporte como un medio en el que el estudiante se sienta y comporte como universitario en el plano de su disfrute cultural e ideológico, con un alto nivel de sensibilidad y de identidad con su centro.

El proceso de la DEU está vinculado a los demás procesos clave que realiza la universidad. A pesar de estar relacionado directamente con la promoción de la cultura y el deporte; de una forma u otra está relacionado con la formación e investigación por la gran cantidad de procesos que abarca: gestión del movimiento deportivo, gestión del movimiento de artistas aficionados, gestión de las cátedras honoríficas, gestión de los cursos de capacitación, gestión de las instituciones culturales y gestión de los proyectos socioculturales. Dichos procesos se relacionan entre sí y tienen el fin común de promover cultura donde las entradas de un subproceso pueden ser las salidas de otro. Preparando a los futuros profesionales con una cultura general más amplia, sinónimo de formación integral, de desarrollo político-ideológico, de competencia profesional, de incondicionalidad y de defensa de la Revolución en el campo de las ideas enmarcadas al deporte.

1.1.3. Movimiento Deportivo

Uno de los procesos más abarcadores dentro de la EU en la UCI es el movimiento deportivo, este integrado por estudiantes y profesores que trabajan en conjunto en clases de educación física, entrenamientos y preparación deportiva en las diferentes disciplinas del deporte como lo son el beisbol, futbol sala y 11, atletismo,

gimnasia, voleibol, maratones entre otros. Este esfuerzo se encuentra reflejado en el momento en que los participantes entregan su máxima dedicación para lograr buenos resultados dentro de los Juegos Mella.
Los juegos constituyen el momento cumbre del Movimiento Deportivo, que se realiza de carácter anual en eventos a nivel de universidad y provinciales, así como eventos de carácter provincial y nacional cada dos años. El Movimiento Deportivo es reconocido por el Ministerio de Instituto Nacional de Deporte, Educación Física y Recreación (INDER), como uno de los de mayor nivel y calidad en el país, avalado por su sistematicidad y proyección social (Espronceda Pérez; Pérez Fuentes y Remedio Frometa, 2021). Para un deportista poder participar en esta actividad en la UCI, debe presentar una planilla de inscripción con un conjunto de datos firmada por el vicerrector de extensión y residencia del mismo, como: nombre y apellidos, carrera, facultad y si es estudiante egresado.

Los juegos deportivos se realizan anualmente en todas las universidades del país. Comienzan desde la base que son los juegos deportivos a nivel de facultad, luego se realizan a nivel de universidad, en caso de haber más de una universidad en la provincia se realizan los juegos deportivos a nivel provincial y termina a nivel nacional. Entre cada evento se realiza un proceso de selección con los ganadores de su disciplina del nivel inferior inmediato con el fin de conformar un conjunto de los atletas más destacados y con mejor participación para garantizar la calidad de los juegos deportivos en un nivel más selectivo como lo son los juegos provinciales y nacionales.
Los juegos deportivos nacionales de la FEU es considerado como la bienal deportiva de la Educación Superior cubana, constituye el máximo peldaño de un ciclo competitivo que abarca dos cursos académicos. En estos se premia el esfuerzo y la dedicación de aficionados, profesores, directivos y los entrenadores, sin los cuales no fuera posible exhibir el prestigio que le ha merecido el respaldo incondicional del INDER y su red de consejos, instituciones y organizaciones deportivas (MES, 2022).

1.1.4. Gestión de información

Gestión de la información: es la denominación convencional de un conjunto de procesos por los cuales se controla el ciclo de vida de la información, desde su obtención (por creación o captura), hasta su disposición final (su archivo o eliminación). Tales procesos también comprenden la extracción, combinación, depuración y distribución de la información a los interesados.
El objetivo de la Gestión de la Información (GI) es garantizar la integridad, disponibilidad y confidencialidad de la información (Barzaga-Sablón; Pincay; Nevárez-Barberán y Cobeña, 2019). Según la revista informática-tecnológica cubana de Joven club la GI es el proceso de controlar, almacenar y recuperar la información adquirida por una entidad, a través de diferentes fuentes. De igual modo pone en uso los recursos de información de la organización, de origen interna como externa para operar, aprender y adaptarse a los cambios del ambiente (JovenClub, 2020). Para el autor del trabajo la gestión de información es el control, almacenamiento y recuperación de información por parte de una entidad para garantizar la integridad, disponibilidad y confidencialidad de la misma.

Los sistemas de gestión de información se basan en la recopilación y presentación de datos. Su propósito es hacer que la toma de decisiones por parte de los especialistas sea más eficiente y productiva. Por lo tanto, un sistema de gestión de la información puede considerarse como un instrumento que permite la gestión de los recursos de información de una organización. Su finalidad es generar servicios que respondan a las necesidades de los usuarios, aprovechando al máximo sus recursos en función de la mejora continua y de la toma de decisiones (M. D. Pérez, 2017).

1.2. Estudio de sistemas homólogos

Actualmente existen diversas soluciones para la gestión de procesos. Pero en el ámbito de la presente investigación, se tomaron en cuenta solamente aquellas que permitieran la gestión de procesos de extensión. A continuación, se relacionan sus características y principales funcionalidades.

1.2.1. Ámbito internacional

- **Sistema Nacional de Información de la Educación Superior (SNIES):** Es un sistema de información creado para responder a las necesidades de recopilar información de la Educación Superior en Colombia. Este sistema como fuente de información, en relación con las instituciones y programas académicos aprobados por el Ministerio de Educación Nacional, consolida y suministra datos, estadísticas e indicadores. Por la consolidación de información que hace el SNIES de las diferentes entidades del sector, es posible darle a conocer datos de interés a la comunidad en general sobre los avances en la Educación Superior colombiana (Educación Nacional de Colombia, s.f.). Este sistema no es factible como solución de la presente investigación debido a que maneja información generada por instituciones de educación superior, pero no maneja información relacionada con el Movimiento Deportivo. Además, se gestiona la información de diferentes formas incluyendo en formato duro, siendo esto una de las problemáticas a resolver por la presente investigación, sin embargo, es válido considerar que de la manera en la que muestra la información puede ser referencia para el módulo a desarrollar.

1.2.2. Ámbito nacional

- **Sistema informático de gestión universitaria:** Esta aplicación web fue diseñada con el objetivo final de facilitar el trabajo en los diferentes departamentos y secretarías docentes, al igual que a los estudiantes en el Instituto Superior de Arte (Instituto Superior de Arte (ISA)). La interfaz Web que brinda esta aplicación permite agregar nuevos datos y realizar consultas a los mismos. La información introducida se almacena en la base de datos modelada para este fin (Jiménez[1], s.f.). Dicho sistema no cuenta con una herramienta o sección para gestionar la información asociada con el movimiento deportivo, a pesar de ello, brinda una interfaz grádica bastante intuitiva y acorde al hambiente de rasgos

universitarios.

- **Página web extensión universitaria de la Universidad de Holguín:** Esta página web facilita la integración y la divulgación de actividades y es de fácil acceso para el público. La cual tiene como objetivo mantener informados al estudiantado y al claustro docente. También se muestran los resultados alcanzados hasta la actualidad (Quiñones Laffita; Hernández Torres y Cordón González, 2018). Este sistema gestiona información acerca de las áreas docentes, productivas e investigativas de la universidad con respecto a la Residencia, Cooperación, Biblioteca, Investigación, Tecnologías, Tele Formación, Ingreso y Egreso, así como Extensión, Pregrado y Postgrado. El cual fue desarrollado por la Dirección de Informatización perteneciente a la Vicerrectoría de la Universidad de las Ciencias Informáticas, y aunque no se ajusta como respuesta a la situación problemática de la presente investigación, se toma como referencia por la forma en la que gestiona la información.

- **Sistema de Gestión Universitaria Sistema de Gestión Universitaria (SIGENU):** Es un sistema informático distribuido del tipo cliente-servidor, construido sobre la base de la utilización de software libre, con el uso de Java como lenguaje de programación. Surge en el año 2004 como una iniciativa del MES. Constituye la principal línea de desarrollo para la informatización de la GI en la Educación Superior en Cuba. Actualmente el SIGENU cuenta con una base de datos de más de 120 tablas donde se almacenan todos los datos necesarios para el funcionamiento del sistema. En estos momentos los módulos de Matrícula, Plan de Estudios, Control de Estudiantes, Administración, Seguridad, Estadísticas-CES, Toma de decisiones, así como el Almacén de Datos (Datawarehouse) del Sistema, se encuentran implantados en casi todos los Centros de Educación Superior Centros de Educación Superior (CES) adscriptos al MES, así como en el Instituto Superior de Diseño Industrial Instituto Superior de Diseño Industrial (ISDI) y en el Instituto Superior de Relaciones Internacionales (Instituto Superior de Relaciones Internacionales (ISRI)), el Fajardo y otros centros no adscriptos alMES. Todos los CES del país disponen de su matrícula de los cursos diurnos Cursos Diurnos (CD) y cursos por encuentro Curso por Encuentros (CPE) actualizada. Los servicios implantados incluyen un Almacén de Datos a nivel de MES que posibilita la integración de la información de matrícula de todos los CES, tanto adscriptos al MES, como no adscriptos. Esta aplicación informática almacena la información de matrícula de más de 400 mil estudiantes, y brinda un eficiente servicio de reportes fácilmente configurables, además de un servicio de búsqueda detallada de estudiantes por sus distintas características (Lozano, 2018). La efectividad que presenta éste sistema para ser tomado como referencia está en la manera en la que realiza los reportes estadísticos y el tratamiento de la información a partir de gráficas y tablas, aún así cuando no cuenta con la gestión de los procesos extensionistas como tal.

- **Solución para la gestión de información de la Extensión Universitaria en el área de Cultura Física en la Universidad de las Ciencias Informáticas:** Dicho sistema es capaz de mostrar la infor-

mación derivada de la creación de nuevas ediciones de Juegos Deportivos en la UCI. Las competencias son agrupadas, atendiendo a sus características, según el evento deportivo en cuestión. Se tiene como ejemplo de que todas las competencias poseen criterios de búsquedas que facilitarán el acceso a la información, siendo común la selección de la edición deportiva que se desea consultar. En el caso de las competencias que incluyen deportes con pelota se brinda la posibilidad de organizar la búsqueda utilizando los criterios como sexo, grupo deportivo y fecha, mostrando así los enfrentamientos por grupos de las distintas facultades. En el caso de las competencias relacionadas a los deportes de combate, el sistema muestra los resultados de los diferentes eventos según la división del atleta. Las competencias por modalidades, igualmente brindan la posibilidad de organizar la búsqueda según el interés del usuario. Por último, este sistema permite la posibilidad de mostrar otros tipos de competencias, estas son las que no presentan un sistema competitivo definido, por cuanto solo interesa visualizar los resultados finales de las mismas (Gómez Hernández; Céspedes Lara; Llano Castro; Cruz Amarán y Terrero Galano, 2015). A partir de todas las características que brinda este sistema se puede llegar a la conclusión de que, aunque no gestiona la información de todo el movimiento deportivo en la UCI, pues se limita solo a la informatización de los juegos inter facultades, si es vital para entender el proceso y el tratamiento de la información que generan los eventos deportivos dentro de dicho movimiento en la Universidad.

1.2.3. Consideraciones del estudio de sistemas homólogos

Como se refleja en el estudio realizado a los sistemas anteriores, que de una forma u otra se relaciona con la gestión de información en el proceso extensionista se concluye que no satisfacen los problemas existentes en la Universidad. En la bibliografía consultada no se identifica un sistema que realice la gestión de la información del movimiento deportivo aplicable a la UCI. Por lo tanto, constituye una necesidad primordial para la universidad desarrollar un módulo que permita gestionar dicho proceso.

1.3. Metodologías, herramientas, y tecnologías

La presente solución constituye un módulo para el Sistema Gerencial de Extensión Universitaria, por lo que la Metodologías, herramientas, y tecnologías a emplear son las empleadas en el desarrollo del sistema.
Para guiar el proceso de desarrollo la metodología seleccionada es el AUP en su variante UCI, haciendo uso del escenario 4 para mantener la uniformidad en el proceso de desarrollo del sistema sobre el cual va a estar el módulo a desarrollar.

1.3.1. Modelado de software

- **Lenguaje Unificado de Modelado** 8.0: Lenguaje de Modelado Unificado (UML, por sus siglas en inglés) desempeña un rol importante no solo en el desarrollo de software, sino también en los sistemas que no tienen software en muchas industrias, es una forma de mostrar visualmente el comportamiento

y la estructura de un sistema o proceso. UML ayuda a mostrar defectos potenciales en las estructuras de aplicaciones, el comportamiento del sistema y otros procesos empresariales. Este lenguaje de modelado automatiza, en parte, la producción de software y los procesos, simplifica las complejidades y ayuda a resolver los problemas arquitectónicos constantes (Koç; Erdoğan; Barjakly y Peker, 2021). UML será empleado en la elaboración de los diagramas de la presente investigación.

- **Visual Paradigm 8.0**: Visual Paradigm es una herramienta de Ingeniería de Software Asistida por Computadora Ingeniería de Software Asistida por Computadora (CASE, por sus siglas en inglés) que utiliza el lenguaje de modelado UML y con la cual se pueden elaborar una gran variedad de diagramas para dar soporte al ciclo de vida del desarrollo de software. Es ideal para ingenieros de software, analistas de sistemas y arquitectos de sistemas que están interesados en la construcción de sistemas a gran escala y necesitan confiabilidad y estabilidad en el desarrollo (Paradigm, 2022). Entre sus principales características se encuentran (Pressman, 2005):

  1. Disponibilidad en múltiples plataformas (Windows, Linux).
  2. Capacidades de ingeniería directa e inversa.
  3. Posee licencia gratuita y comercial.
  4. Soporta aplicaciones web.
  5. Permite realizar diagramas de flujo de datos.
  6. Generación de bases de datos. Transformación de diagramas de Entidad-Relación en tablas de base de datos.
  7. Editor de figuras .

- **Balsamiq 4.5.3**: Herramienta de software para la creación de prototipos de interfaces de usuario ("wireframes") de manera rápida y sencilla. Con Balsamiq, los diseñadores pueden crear bocetos de interfaces de usuario y compartirlos con el equipo de desarrollo, clientes u otros interesados en el proyecto (Faranello, 2012). La interfaz de usuario de Balsamiq es muy intuitiva y se asemeja a un bloc de notas con herramientas de dibujo básicas. La aplicación cuenta con una amplia variedad de plantillas de wireframes predefinidas para diferentes tipos de proyectos, como aplicaciones móviles, sitios web o software de escritorio. También permite importar imágenes y elementos gráficos para personalizar el diseño.

1.3.2. Herramientas y tecnologías

Entorno de desarrollo

- **Editor de texto-código Ligero**: Visual Studio Code: Es un Editor de texto-código Ligero que nos permite implementar en distintos lenguajes de programación. Proporciona terminaciones inteligentes basadas en tipos de variables, definiciones de funciones y módulos importados. Permite el código directamente desde el editor. Es compatible con herramientas de control de versiones como Git. Tiene

la capacidad de instalar extensiones adicionales para incorporar nuevas funciones al editor (Microsoft, 2022). Tiene opciones de autocompletado y es muy utilizado en el desarrollo de aplicaciones web. Entre la multitud de lenguajes que soporta se encuentran JavaScript, Python, Lenguaje de marcado de hipertexto (HTML, por sus siglas en inglés), Hoja de estilo en Cascada (CSS, por sus siglas en inglés), Notación de objetos JavaScript (JSON, por sus siglas en inglés), Preprocesador de Hipertexto (PHP, por sus siglas en inglés), entre otros; los cuales son muy usados en el desarrollo de aplicaciones web.

Lenguajes de programación

- **Python 3.10.5**: Python es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el aprendizaje automático. Los desarrolladores utilizan Python porque es eficiente y fácil de aprender, además de que se puede ejecutar en muchas plataformas diferentes. El software Python se puede descargar gratis, se integra bien a todos los tipos de sistemas y aumenta la velocidad del desarrollo. Es uno de los lenguajes más populares actualmente en el desarrollo web del lado del servidor (Van Rossum et al., 2007). Entre sus principales ventajas cabe mencionar (Balogun, 2022):

  1. Cuenta con una gran biblioteca estándar que contiene códigos reutilizables para casi cualquier tarea. De esta manera, los desarrolladores no tienen que escribir el código desde cero.
  2. Se puede utilizar fácilmente con otros lenguajes de programación conocidos.
  3. La comunidad activa de Python incluye millones de desarrolladores alrededor del mundo que prestan su apoyo. Si se presenta un problema, puede obtener soporte rápido de la comunidad.
  4. Se puede trasladar a través de diferentes sistemas operativos de computadora, como Windows, macOS, Linux y Unix.
  5. Hay muchos recursos útiles disponibles en Internet si desea aprender Python.
  6. Permite escribir un programa de Python con menos líneas de código en comparación con muchos otros lenguajes.

- **JavaScript**: JavaScript es un robusto lenguaje de programación que se puede aplicar a un documento HTML y usarse para crear interactividad dinámica en los sitios web. Es el lenguaje de desarrollo web más popular para programar el comportamiento de las páginas web. Permite añadir eventos cuando se presionan botones o cuando son introducidos datos en formularios. Soporta librerías de terceros, permite incorporar funcionalidades de otros proveedores de contenido, es compacto y flexible (J. E. Pérez, 2019).
- **HTML 5**: El HTML es el código que se utiliza para estructurar y desplegar una página web y sus contenidos. HTML no es un lenguaje de programación; es un lenguaje de marcado que define la estructura de tu contenido. En la actualidad todas las páginas web están estructuradas con este lenguaje (Malqui, 2018).
- **CSS** 3: CSS es el lenguaje que se usa para diseñar un documento HTML. Describe cómo deben

mostrarse los elementos HTML. Ahorra mucho trabajo debido a que puede controlar el diseño de varias páginas web a la vez, incluyendo el diseño y las variaciones en la visualización para diferentes dispositivos y tamaños de pantalla (Puig, 2013).

Framework de desarrollo

- **Django 4.0.1**: Django es un marco web Python de alto nivel que fomenta un desarrollo rápido y un diseño limpio y pragmático. Creado por desarrolladores experimentados, se ocupa de gran parte de las molestias del desarrollo web. Es gratis y de código abierto. Es extremadamente escalable. Algunos de los sitios más concurridos de la web aprovechan la capacidad de Django para escalar de forma rápida y flexible. Es uno de los frameworks que más se han estado utilizado en los últimos años para el desarrollo web (D. Rubio, 2017; Vidal-Silva; Sánchez-Ortiz; Serrano y J. M. Rubio, 2021).

Bibliotecas

- **Bootstrap 5**: Bootstrap es una librería de código abierto que permite diseñar rápidamente sitios web adaptables con su sistema de cuadrícula con diseño de respuesta, permitiendo que el sitio sea capaz de visualizarse correctamente en dispositivos con diferentes tamaños de pantalla. Solo se ocupa del desarrollo del front-end, posee gran cantidad de componentes preconstruidos y potentes complementos de JavaScript (Jacob Thornton, 2022; Niska, 2014). Es compatible con la mayoría de navegadores webs actuales.
- **jQuery**: jQuery es una biblioteca multiplataforma de JavaScript rápida, pequeña y rica en funciones. Hace que cosas como el recorrido y la manipulación de documentos HTML, el manejo de eventos, la animación y Asynchronous JavaScript and Xivas Xivas Ascícronos de JavaScript (AJAX, por sus siglas en inglés) sean mucho más simples con una Application Programming Interfaces Progrma de Interfáz de Aplicación (API, por sus siglas en inglés) fácil de usar que funciona en una multitud de navegadores. Con una combinación de versatilidad y extensibilidad (Foundation, 2022). Permite manejar eventos y añadir animaciones. Tiene soporte para AJAX, lo que permite enviar peticiones al servidor y, en base a la respuesta, modificar una parte de la web sin tener la necesidad de recargar toda la página (Ventura Bautista, 2021).

Sistema gestor de base de datos

- **PostgreSQL 14.3**: Es un poderoso sistema de base de datos relacional de objetos de código abierto con más de 30 años de desarrollo activo que le ha valido una sólida reputación por su confiabilidad, robustez de funciones y rendimiento. Utiliza y amplía el Lenguaje de Consultas Estructurado (SQL, por sus siglas en inglés) combinado con muchas características que almacenan y escalan de forma segura las cargas de trabajo de datos más complicadas. Es un software libre de código abierto que se ejecuta en todos los principales sistemas operativos. Posee una amplia comunidad y permite manejar

un gran número de tipos de datos Pilicita Garrido; Borja López y Gutiérrez Constante, 2021.

Herramientas para control de versiones

- **Git**:Para garantizar el proceso de control de versiones se propone usar el cliente Git para la plataforma Gitlab disponible en la UCI (Miller, 2022). Este sistema es de código abierto y gratuito. Entre sus principales características están:

  1. Mantener un único repositorio de código fuente.
  2. Automatizar la construcción del proyecto.
  3. Autodiagnóstico de la construcción.
  4. Entregar los cambios diariamente a la línea base.
  5. Construir la línea base tras cada entrega.
  6. Probar en una réplica del entorno de producción.
  7. Mantener una ejecución rápida de la construcción del proyecto.
  8. Fácil obtención del último ejecutable del proyecto.
  9. Publicar el estado del proyecto.
  10. Automatizar el despliegue.

1.4. Conclusiones del capítulo

En este capítulo se abordaron los elementos teóricos que sustentan la propuesta solución del problema en cuestión. Esto permite arribar a las siguientes conclusiones:

- La gestión de la información se lleva a cabo manualmente, por lo cual se definió como propuesta solución un sistema web que permita a los profesores de la DEU la gestión del movimiento deportivo de manera centralizada y accesible en todo momento.
- El análisis de sistemas homólogos permitió identificar características necesarias en la aplicación y comprender la necesidad de desarrollar un sitio propio que se adapte a las necesidades de la UCI.
- A partir del análisis de las herramientas, tecnologías y metodología con el cual fue desarrollado el Sistema Gerencial de Extensión Universitaria (SGEU) se emplea como metodología de desarrollo AUP variante UCI en su escenario 4, Django como framework de desarrollo, como lenguajes de programación como Python.

CAPÍTULO 2

Análisis y diseño del módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria

El presente capítulo aborda los principales aspectos relacionados con los requisitos de la propuesta de solución. Se identifican los requisitos funcionales y no funcionales con los que debe cumplir la solución propuesta, así como estilo arquitectónico, los diagramas de clases, modelo de datos y los patrones de diseño para lograr buenas prácticas en el diseño y posterior implementación del sistema. Igualmente se muestran los principales artefactos de ingeniería de software propuestos por la metodología utilizada.

2.1. Descripción de la propuesta de solución

A partir del problema y el objetivo planteado en la presente investigación, se propone como solución un módulo para la gestión de la información asociada al Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria. El software se desarrollará sobre la base de una aplicación web posibilitándoles a los usuarios disfrutar de los beneficios que estas aplicaciones reportan, el acceso múltiple al sistema por medio de un servidor web, así como las actualizaciones y los mantenimientos del software sin necesidad de distribuir e instalar. La gestión de la información se realizará a través de una base de datos localizada en el servidor lo que permitirá, realizarla de forma simultánea por varios usuarios.
El módulo contribuye a la gestión de la información del Movimiento Deportivo, generada durante el proceso de EU en la UCI dirigido por la DEU. El módulo permite la centralización de la información asociada al Movimiento Deportivo almacenada de manera normalizada, facilitando su búsqueda y visualización. El sistema abarca los siguientes procesos:

Evento deportivo

El sistema permite visualizar todos los eventos deportivos registrados en la base de datos, accesible para todos los usuarios. Si el usuario es del rol encargado y cuenta con los permisos adecuados tendrá acceso a gestionar la información de los deportes.

Delegaciones Deportivas

Permite visualizar los equipos de deporte con sus integrantes registrados en el sistema. Si el usuario es del rol encargado y cuenta con los permisos adecuados tendrá acceso a gestionar la información de los equipos deportivos.

Cronograma Deportivo

En esta sección el usuario puede visualizar el cronograma deportivo según los eventos registrados, éstos se muestran ordenados por fecha en orden ascendente para generar un orden de cronograma. Debe permitir exportar el cronograma en formato pdf.

Deportistas

En esta sección se podrá apreciar un directorio de deportistas con la posibilidad de poder buscar al deportista deseado, así como realizar la gestión de éstos al contar con el rol requerido.

2.2. Especificación de requisitos del sistema

Con el objetivo de que el sistema propuesto se desarrolle con las necesidades dictadas por el cliente final a partir del uso de las entrevistas y encuestas se obtuvieron los requisitos funcionales y no funcionales.

2.2.1. Requisitos funcionales

Teniendo en cuenta las características de la propuesta en cuanto a gestión y centralización de la información se identifican 27 requisitos funcionales. A continuación, la tabla 2.1 muestra los requisitos del sistema.

Tabla 2.1. Descripción de los Requisitos Funcionales. (Fuente: Elaboración propia).

No.	Nombre	Prioridad	Complejidad
RF1	Crear Delegación Deportiva	Alta	Media
RF2	Listar Delegación Deportiva	Baja	Baja
RF3	Modificar Delegación Deportiva	Media	Baja
RF4	Mostrar Delegación Deportiva	Media	Baja
RF5	Eliminar Delegación Deportiva	Baja	Media
RF6	Buscar Delegación Deportiva	Media	Baja
RF7	Asignar Premio a la Delegación Deportiva	Alta	Baja
RF8	Modificar Premio a la Delegación Deportiva	Alta	Media
RF9	Eliminar Premio de la Delegación Deportiva	Alta	Media
RF10	Crear Evento Deportivo	Alta	Baja
RF11	Listar Evento Deportivo	Alta	Media
			Continúa en la siguiente página

No.	Nombre	Prioridad	Complejidad
RF12	Modificar Evento Deportivo	Alta	Baja
RF13	Eliminar Evento Deportivo	Alta	Baja
RF14	Mostrar detalles del Evento Deportivo	Alta	Baja
RF15	Buscar Evento Deportivo	Alta	Baja
RF16	Insertar evidencia (Multimedia) del Evento Deportivo	Media	Baja
RF17	Eliminar evidencia (Multimedia) del Evento Deportivo	Alta	Media
RF18	Mostrar evidencia (Multimedia) del Evento Deportivo	Media	Baja
RF19	Mostrar Cronograma deportivo	Alta	Media
RF20	Buscar Evento específico en el cronograma	Alta	Media
RF21	Generar PDF del Cronograma deportivo	Alta	Media
RF22	Crear Deportista	Alta	Media
RF23	Listar Deportista	Alta	Media
RF24	Modificar Deportista	Alta	Media
RF25	Eliminar Deportista	Alta	Media
RF26	Buscar Deportista	Alta	Media
RF27	Mostrar detalles de los Deportistas	Alta	Media

2.2.2. Requisitos no funcionales

- Usabilidad.
  - RnF1. Tipo de aplicación informática: el módulo debe ser web, que contemple estándares y patrones de diseño de Interfaz de Usuario (UI, por sus siglas en inglés) modernas.
  - RnF2. El sistema debe poseer una UI fácil de utilizar para cualquier tipo de usuario con conocimientos básicos de computación en el manejo de ordenadores.
- Confiabilidad.
  - RnF3. El sistema debe ser tolerante a fallos, y mostrar solo la información necesaria para orientar al usuario.
- Hardware y software requerido para utilizar la aplicación.
  - RnF4. Sistema Operativo: Podrá utilizar cualquier sistema operativo.
  - RnF5. El sistema debe de ejecutarse sobre cualquier navegador web con versiones en soporte en adelante.
  - RnF6. La computadora que se utilice como cliente debe contar como mínimo con 2 GB de RAM y la que se utilice para el servidor debe contar como mínimo 500 GB de almacenamiento y como mínimo 4 GB de RAM.
- Eficiencia del desempeño.
  - RnF7. El sistema debe permitir que los usuarios (100) interactúen con él de manera concurrente.
  - RnF8. El tiempo de demora de una petición al servidor debe ser menor a dos (2) segundos

- Seguridad.
  - RnF9. El acceso a la gestión de la información debe estar restringido por usuario, contraseña y rol.
  - RnF10. Se podrá acceder a información sin necesidad de autenticarse.
- Restricciones de diseño e implementación.
  - RnF12. El marco de trabajo que se utilizará está basado principalmente en el uso de herramientas no privativas:
    - Lenguaje de programación del lado del servidor Pyhton v3.10.5.
    - Lenguaje de programación del lado del cliente JavaScript ECMASCRIPT2015.
    - Lenguaje para el estilo del documento web Hoja de estilo en Cascada Versión 3 (CSS3, por sus siglas en inglés).
    - Lenguaje para la estructuracón del documento web Leguaje de marcado de hipertexto Versión 5 (HTML5, por sus siglas en inglés).
    - Lenguaje para el modelado del proceso UML v2.5.1.
    - Framework de desarrollo web Django v4.0.1
    - Framework para los estilos Bootstrap v5.1.
    - Herramienta para el modelado del proceso Visual Paradigm v15.1 .
    - Herramienta para la codificación Microsoft Visual Studio Code v1.68.
    - Herramienta para la realización de pruebas de rendimiento Locust v2.13.0 .
    - Herramienta para la realización de pruebas de seguridad Acunetix Web Vulnerability Scanner v9.5.
    - Herramienta para la automatización de pruebas funcionales Selenium IDE v3.17.2.

2.3. Descripción de las historias de usuario

Para el encapsulamiento de los requisitos del software, tal como lo plantea la metodología Proceso Unificado Ágil versión UCI (AUP-UCI, por sus siglas en inglés), en su escenario 4, se generaron un total de 27 Historias de Usuario (HU), a continuación, se muestran las correspondientes a los requisitos funcionales Crear Delegación Deportiva y Registrar Deportista , el resto se encuentra en el anexo .3:

Tabla 2.2. Historia de usuario # 1

Historia de usuario	
Número: 1	**Nombre:** Crear Delegación Deportiva
Usuario: Usuario común	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** No. 1

Continúa en la próxima página

Tabla 2.2. Continuación de la página anterior

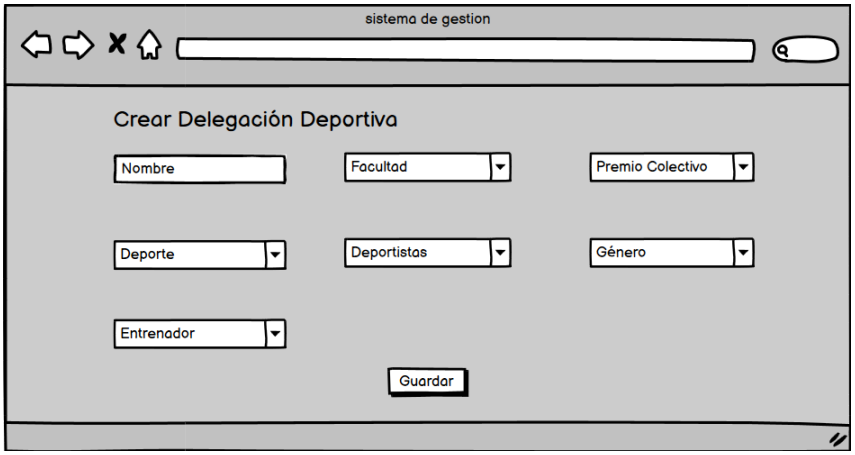
Programador responsable: Airely Sanchez Miralles
Descripción: : El sistema debe permitir crear una Delegación Deportiva a través de un formulario donde el usuario rellena todos los campos. Para ello debe estar autenticado y contar con el rol requerido. • Nombre: Campo de texto (admite caracteres especiales y números). • Deporte: Campo de Selección. • Género: Campo de selección. • Facultad: Campo de selección. • Premio Colectivo: Campo de selección. • Entrenador: Campo de selección. • Deportistas: Campo de selección.
Observaciones: La UI solo aparecerá si el usuario está autenticado. No se pueden dejar campos vacíos y los campos nombre y entrenador deben ser escritos con letra inicial mayúscula, en caso contrario el sistema debe mostrar un mensaje de error que avise al usuario el tipo de error y el lugar dónde se cometió.
Interfaz: 

Tabla 2.3. Historia de usuario # 2

Historia de usuario	
Número: 2	**Nombre:** Registrar Deportista
Usuario: Responsable de la actividad	
Prioridad en negocio: Prioridad	**Riesgo en desarrollo:** Alta
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	

Continúa en la próxima página

Tabla 2.3. Continuación de la página anterior

Descripción: El sistema debe permitir agregar un nuevo Deportista a través de un formulario donde el usuario rellena todos los campos.

- Nombre y Apellidos: Campo de texto.Admite caracteres especiales, letras y números.
- Edad: Campo de selección numérico, admite del 1 al 100.
- Estado: Campo de selección.
- Facultad: Campo de selección.
- Deporte: Muestra el listado existente y el Campo de selección para añadir deportes.
- Género: Campo de selección.
- Medalla: Campo de selección.

Observaciones: Para ello debe estar autenticado y contar con el rol requerido, en el campo de Nombre y Apellidos se deberá comenzar a escribir con letras Mayúsculas en caso contrario se lanzará un error al igual que si se deja en blanco el campo.

Interfaz:

2.4. Estilo arquitectónico

El diseño arquitectónico se interesa por entender cómo debe organizarse un sistema y cómo tiene que diseñarse la estructura global de ese sistema. Es la primera etapa en el proceso de diseño del software. Es el enlace crucial entre el diseño y la ingeniería de requerimientos. Identifica los principales componentes estructurales en un sistema y la relación entre ellos. La salida del proceso de diseño arquitectónico consiste en un modelo arquitectónico que describe la forma en que se organiza el sistema como un conjunto de componentes en comunicación (Sommerville, 2011).

Para el diseño de la propuesta de solución se utiliza el estilo Llamada y Retorno. Este estilo permite que los datos sean pasados como parámetros y el manejador principal proporciona un ciclo de control sobre las subrutinas.

2.4.1. Patrón arquitectónico

Un patrón arquitectónico se puede considerar como una descripción abstracta estilizada de buena práctica, que se ensayó y puso a prueba en diferentes sistemas y entornos. De este modo, un patrón arquitectónico debe describir una organización de sistema que ha tenido éxito en sistemas previos. Debe incluir información sobre cuándo es y cuándo no es adecuado usar dicho patrón, así como sobre las fortalezas y debilidades del patrón (Sommerville, 2011).

Como propuesta de solución para el desarrollo de la aplicación se determinó el patrón Modelo Vista Plantilla (MTV, por sus siglas en inglés) que propone Django como una versión del Modelo Vista Controlador (MVC, por sus siglas en inglés). La figura 2.1 muestra el diagrama de paquetes de la propuesta de solución siguiendo el patrón MTV.

El funcionamiento de este estilo comienza cuando el usuario a través del navegador envía la solicitud a la Vista. La vista se encarga de ejecutar la lógica empresarial e interactuar con un modelo para transportar datos y representar una plantilla. El modelo ayuda a manejar la base de datos. Es una capa de acceso a datos que maneja los datos. La plantilla es una capa de presentación que maneja completamente la parte de la interfaz de usuario. La vista se utiliza para ejecutar la lógica empresarial e interactuar con un modelo para transportar datos y representar una plantilla (Jaiswal, 2016).

Figura 2.1. Representación de la arquitectura.

- **Modelo:** Es la parte del MTV que hace de medio de almacenamiento de datos, donde se puede indicar y controlar el comportamiento de los datos. Representa la información con la que trabaja la aplicación. Este no tiene conocimiento específico de las vistas o de las plantillas, ni siquiera contiene referencias a ellos.
- **Vistas:** Contiene la lógica de la aplicación, la que responde a las peticiones de los usuarios e interactúa con la plantilla y hace uso del modelo de datos para poder acceder a la información almacenada. Es el equivalente del controlador en MVC. Es necesaria para devolver una respuesta hacia el cliente que la solicita, también procesa las peticiones o solicitudes que accederán al modelo para poder entregar u obtener los datos.
- **Plantillas:** Es la interfaz de usuario, es la parte del Framework que permite mostrar los datos almacenados y procesados. Da la posibilidad al usuario de interactuar con el sistema. Decide la forma en

la que se presentarán los datos devueltos por la vista en el navegador web utilizando estilos CSS o brindando dinamismo a través de JavaScript.

2.5. Diagrama de clases de diseño

Un diagrama de clase es un tipo de diagrama UML donde se describe el sistema a través de los diferentes tipos de objetos que lo componen, las relaciones entre ellos, sus métodos y atributos. Al aplicar estereotipos web se describen los componentes que conforman el modelo, la vista y plantillas presentes en la propuesta solución. Fueron generados en total 10 diagramas de clase. A continuación, la figura 2.2 presenta el diagrama correspondiente a la gestión de las Delegaciones Deportivas.

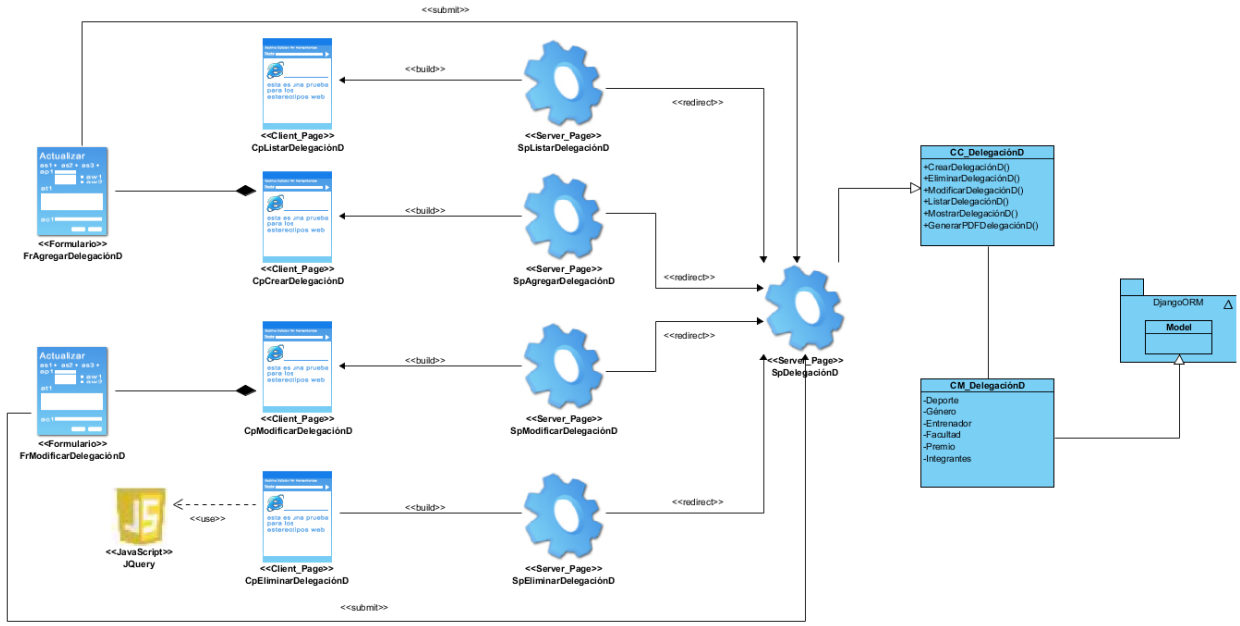


Figura 2.2. Diagrama de clases de gestión de Delegaciones Deportivas

Descripción de las clases

- FrAgregarDelegaciónD: Formulario que contiene la información para la creación de una Delegación Deportiva.
- FrModificarDelegaciónD: Formulario que contiene la información para la modificación de una Delegación Deportiva.
- CpAgregarDelegaciónD: Página HTML donde se encuentra el formulario para la creación de una Delegación Deportiva. Usa JavaScript para realizar validaciones.
- CpModificarDelegaciónD: Página HTML donde se encuentra el formulario para la modificación de una Delegación Deportiva. Usa JavaScript para realizar validaciones.
- CpEliminarDelegaciónD: Página HTML para eliminar una Delegación Deportiva.

- CpListarDelegaciónD: Página HTML donde se listan las Delegación Deportiva.
- SpAgregarDelegaciónD: Página del servidor encargada de construir la página cliente AgregarDelegaciónD.
- SpModificarDelegaciónD: Página del servidor encargada de construir la página cliente ModificarDelegaciónD.
- SpListarDelegaciónD: Página del servidor encargada de construir la página cliente ListarDelegaciónD.
- SpEliminarDelegaciónD: Página del servidor encargada de construir la página cliente EliminarDelegaciónD.
- SpDelegaciónD: Página del servidor encargada de manejar las Delegaciones Deportivas.
- CCDelegaciónD: Clase Controladora donde se implementan todos los métodos de las Delegaciones Deportivas. Se encarga de manipular los modelos.
- CMDelegaciónD: Modelo donde se guardan todos los datos del festival que se almacenan en la base de datos. A su vez son manipulados por el Mapeador de Objeto Relacional (ORM, por sus siglas en inglés) de Djamgo.

El diagrama correspondiente a la gestión de las Delegaciones Deportivas se basa en formularios, páginas HTML, páginas del servidor, una clase controladora y un modelo, todos ellos trabajando en conjunto para guardar, modificar, listar y eliminar información relacionada con las delegaciones deportivas de manera eficiente y estructurada.

2.6. Patrones de diseño

Un patrón de diseño es una solución a un problema de diseño que puede ser utilizada de forma repetida, cuya efectividad ya está probada. Puede ser usado en diferentes contextos, no es código de programación sino una descripción de cómo resolver un problema dado. Con el uso de ellos en la confección de una aplicación, se garantizarán diseños flexibles, reusables y modulares. Cada patrón resuelve un problema en específico, por lo que es importante conocerlos y aplicarlos en la situación en la que lo requiera. Son soluciones bien detalladas que promueven buenas prácticas, facilitando su comprensión y reusabilidad (Gamma; Helm; R. Johnson; R. E. Johnson; Vlissides et al., 1995; Pressman, 2005).

1. Patrones Generales de Software de Asignación de Responsabilidad (GRASP, por sus siglas en inglés): Los patrones GRASP, más que patrones propiamente dichos, son una serie de buenas prácticas de aplicación recomendable en el diseño de software. Entre ellos, los patrones Experto, Creador, Bajo acoplamiento y Alta cohesión guardan directa relación con la creación y asignación de responsabilidades a los objetos (Tabares, 2010).

    - **Experto:** Posibilita una adecuada asignación de responsabilidades facilitando la comprensión del sistema, su mantenimiento y adaptación a los cambios con reutilización de componentes.

Trae como beneficios que se mantiene el encapsulamiento de la información puesto que los objetos usan su propia información para llevar a cabo tareas, lo que permite crear sistemas más robustos y mantenibles, se distribuye el comportamiento de las clases que contienen la información requerida, por lo que las clases se vuelven más fáciles de entender y mantener (Larman, 2003).

- **Creador:** Aporta un principio general para la creación de objetos, una de las actividades más frecuentes en programación. Asigna a una clase la responsabilidad de crear instancias de la otra cuando se cumplen determinadas condiciones en su relación, como son que la primera agrega a la segunda, o contiene los datos de inicialización de la segunda o utiliza estrechamente a esa clase para su funcionamiento (ibíd.).
- **Bajo acoplamiento:** Es una medida de la fuerza con que una clase se relaciona con otras, porque las conoce y recurre a ellas; una clase con bajo acoplamiento no depende de muchas otras, mientras que otra con alto acoplamiento presenta varios inconvenientes; es difícil entender cuando está aislada, es ardua de reutilizar porque requiere la presencia de otras clases con las que esté conectada y es cambiante a nivel local cuando se modifican las clases afines (Tabares, 2010).
- **Alta cohesión:** Es una medida que determina cuán relacionadas y adecuadas están las responsabilidades de una clase, de manera que no realice un trabajo colosal; una clase con baja cohesión realiza un trabajo excesivo, haciéndola difícil de comprender, reutilizar y conservar. Este patrón expresa que se debe asignar a las clases responsabilidades que trabajen sobre una misma área de la aplicación y que no tengan mucha complejidad (ibíd.).
- **Controlador:** Le asigna a una clase la responsabilidad de manejar las respuestas del sistema a las acciones de los usuarios. El controlador define el método para la operación del sistema. Permite aumentar el potencial para la reutilización, asegura que la lógica de la aplicación no se maneje en la interfaz y permite validar las operaciones de manera eficiente y segura (Larman, 2003).

2. Pandilla de cuatro (GOF, por sus siglas en inglés): Son patrones usualmente aplicados en el trabajo orientado a objetos. Los mismos se clasifican de acuerdo a sus funciones en tres grupos: creacionales, estructurales y de comportamiento. De ellos se utiliza:

   - **Constructor:** Separa la creación de un objeto de su representación, para que a través del mismo proceso de creación se obtengan distintas representaciones de un objeto. Se usa para permitir la creación de una variedad de objetos complejos desde un objeto fuente, el objeto fuente se compone de una variedad de partes que contribuyen individualmente a la creación de cada objeto complejo. Tiene como ventajas reducir el acoplamiento y variar la representación interna de estructuras complejas, manteniendo una interfaz común desde el objeto fuente (Hunt, 2013).
   - **Estrategy:** Patrón que consiste en definir una familia de algoritmos, encapsularlos los unos a los otros y hacerlos intercambiables. Se clasifica como patrón de comportamiento porque determina cómo se debe realizar el intercambio de mensajes entre diferentes objetos para resolver una

tarea Permite a los algoritmos variar en dependencia del cliente que los use. Se usa mediante el polimorfismo que caracteriza a las clases en la Programación Orientada a Objeto (POO), otorgándole flexibilidad y mantenibilidad al código (Hunt, 2013).

2.7. Modelo de datos

Para la determinación de una estructura lógica de una base de datos y el modo de almacenar, organizar y manipular los datos se emplea el modelo de datos (Pressman, 2005). Con el objetivo de definir las clases persistentes se identifican los conceptos, en el dominio del negocio, que persisten en el tiempo. A partir del diagrama de clases persistentes se generó el siguiente diagrama entidad-relación, el cual posee un conjunto de tablas correspondientes a cada componente:

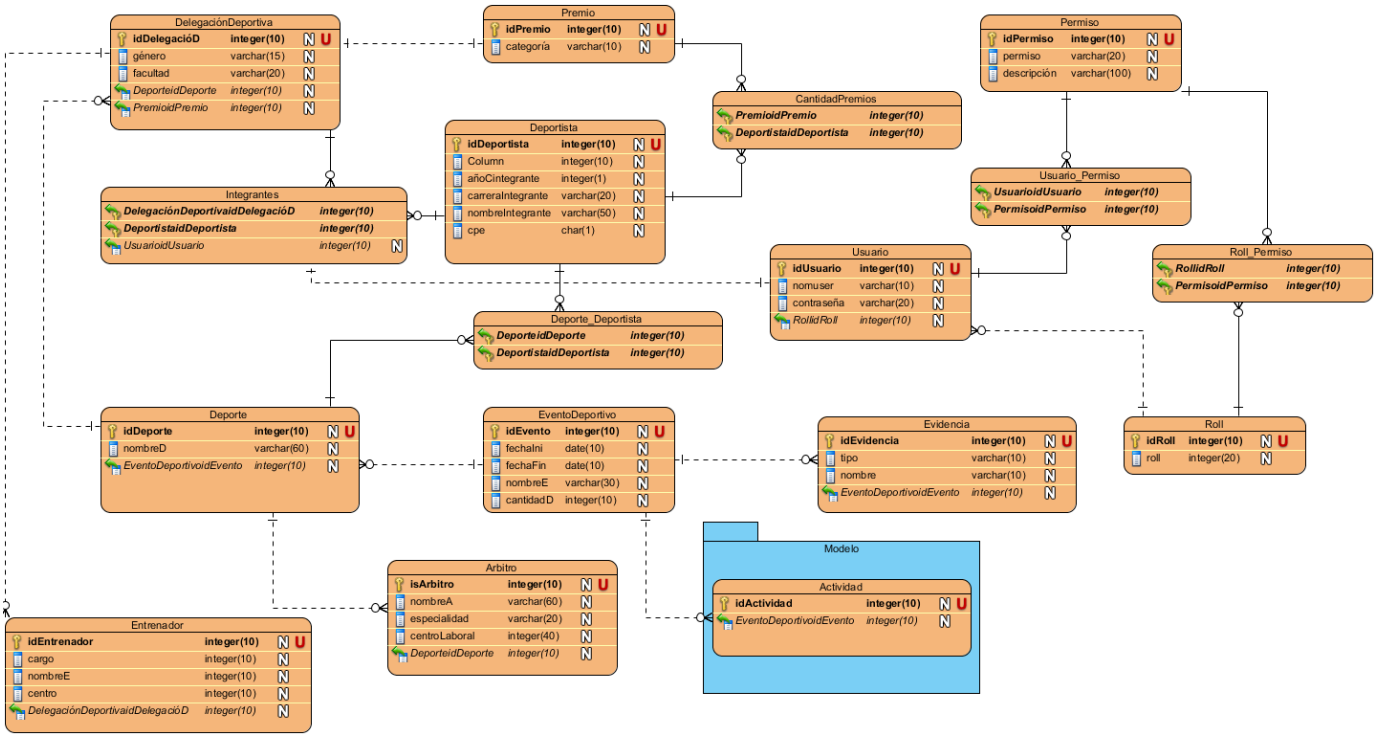


Figura 2.3. Modelo de datos

Descripción de las tablas

1. **Actividad**: Tabla principal del modelo de datos en la que se registran las actividades con sus características.
2. **DelegaciónDeportiva**: Tabla que recoge los datos de las delegaciones deportivas registradas.
3. **Premio**: Tabla donde se registran los datos de los premios.

4. **CantidadPremios**: Tabla nomencladora de los premios de los deportistas donde un deportista puede tener varios premios, y un premio ser asignado a varios deportistas.
5. **Deportista**: Tabla donde se recogen los datos de los deportistas registrados.
6. **Integrantes**: Tabla nomencladora de los integrantes de las delegaciones deportivas, donde una delegación puede tener varios artistas y un artista puede estar vinculado a más de una delegación.
7. **Deporte**: Tabla donde se recopilan los datos de los deportes a efectuarse.
8. **Deporte-Deportista**: Tabla nomencladora de los deportistas a los respectivos deportes en los que están inscritos, donde cada deporte puede tener inscrito a varios deportistas y un deportista puede estar vinculado a más de un deporte.
9. **Usuario**: Tabla que guarda los datos de los usuarios que se autentifiquen en el sistema.
10. **Permiso**: Tabla para recopilar la información asociada a los permisos en la configuración del sistema.
11. **Usuario-Permiso**: Tabla nomencladora de los usuarios y los permisos que les corresponden, donde cada usuario puede poseer varios permisos y un permiso específico puede estar otorgado a varios usuarios.
12. **Rol**: Tabla para guardar los roles del sistema.
13. **Rol-Permisos**: Tabla para establecer la relación de los permisos que posee cada roll, donde un roll puede tener varios permisos y a su vez, un permiso puede pertenecer a varios roles.
14. **EventoDeportivo**: Tabla que recoge los datos de los eventos deportivos a desarrollarse.
15. **Evidencia**: Tabla que guarda los datos de las evidencias generadas a partir de los eventos deportivos.
16. **Arbitro**: Tabla que guarda todos los datos registrados de los árbitros que participan en el desarrollo de los deportes.
17. **Entrenador**: Tabla que recoge los datos de los entrenadores registrados que van a estar vinculados a las delegaciones deportivas.

El modelo de datos consiste en varias tablas que recopilan información sobre las entodades referentes al módulo. Además, se utilizan tablas nomencladoras para establecer relaciones entre las entidades, como los premios asignados a los deportistas, los deportistas inscritos en diferentes delegaciones deportivas, los permisos asignados a los usuarios y los permisos asociados a los roles. Este modelo de datos proporciona una estructura organizada para gestionar y almacenar la información relacionada con el movimiento deportivo para el Sistema Gerencial de Extensión Universitaria.

2.8. Consideraciones parciales

Luego de haber realizado el análisis y diseño del sistema propuesto y generar los artefactos que dispone la metodología seleccionada, se pudo concluir lo siguiente:

- La descripción de la propuesta de solución a través del análisis de sus principales características, permitió identificar las funcionalidades del sistema propuesto, acorde a las necesidades del cliente para contribuir a la gestión del movimiento deportivo en la UCI.

- La especificación de los requisitos del sistema permitió describir cada uno de los requisitos funcionales (27) encapsulados en las HU, así como identificar los requisitos no funcionales (12) teniendo en cuenta un conjunto de características generales y restricciones del mismo, tales como usabilidad, confiabilidad, hadware y software, eficiencia del desempeño, seguridad y restricciones del diseño e implementación.
- El establecimiento de la arquitectura del sistema a partir del patrón arquitectónico, facilitó aislarse del problema en que se enmarca la presente investigación, asegurando un entendimiento común entre las partes interesadas.
- La identificación de los patrones de diseño permitió disminuir el impacto de los cambios futuros en el código fuente de la misma.
- Las características abordadas sobre el diseño del sistema a través de la comprensión de cada elemento que lo componen, facilitó el enfoque en cuanto a composición lógica y física de la propuesta de solución.

CAPÍTULO 3

Implementación y pruebas del módulo del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria

En el presente capítulo se abordan los aspectos relacionados con la implementación de la solución propuesta. Se definen los estándares de codificación para lograr una mejor comprensión y estandarización del código resultante. Se realiza la validación del correcto funcionamiento del software mediante la aplicación de las pruebas funcionales y no funcionales y se presenta el nivel de satisfacción de los usuarios de la solución implementada aplicando la técnica de Iadov.

3.1. Estándares de codificación

Las convenciones o estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código. Un estándar de codificación completo comprende todos los aspectos de la generación de código. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez. Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento.
A continuación, se define el estándar de codificación empleado en el módulo.

Tabla 3.1. Estándares de codificación a utilizar en la implementación del sistema. (Fuente: Elaboración propia).

Tipo de estándar	Descripción
Organización del código	• El código en una página se organizará por bloques • La indentación se realizará solamente con tabulaciones, no debe utilizarse nunca los cuatro (4) espacios
	Continúa en la siguiente página

Tipo de estándar	**Descripción**
Máxima longitud de las líneas	• Todas las líneas deben estar limitadas a un máximo de setenta y nueve caracteres • Dentro de paréntesis, corchetes o llaves se puede utilizar la continuación implícita para cortar las líneas largas.
Líneas en blanco	• Separar las funciones de alto nivel y definiciones de clases con dos (2) líneas en blanco • Las definiciones de métodos dentro de una clase deben separarse por una (1) línea en blanco • Se puede utilizar líneas en blanco escasamente para separar secciones lógicas

Importación	• Las importaciones deben estar en líneas separadas
	• Siempre deben colocarse al comienzo del archivo
	• Deben quedar agrupadas de la siguiente forma:
	  ◦ Importaciones de la librería estándar
	  ◦ Importaciones terceras relacionadas
	  ◦ Importaciones locales de la aplicación / librerías
	• Cada grupo de importaciones debe estar separado por una línea en blanco

	Continúa en la siguiente página

Tipo de estándar	**Descripción**
Espacios en blanco en expresiones y sentencias	• Evitar utilizar espacios en blanco en las siguientes situaciones: ◦ Inmediatamente dentro de paréntesis, corchetes y llaves ◦ Inmediatamente antes de una coma, un punto y coma o dos puntos ◦ Inmediatamente antes del paréntesis que comienza la lista de argumentos en la llamada a una función ◦ Inmediatamente antes de un corchete que empieza una indexación ◦ Más de un espacio alrededor de un operador de asignación (u otro) para alinearlo con otro • Deben rodearse con exactamente un espacio los siguientes operadores binarios: ◦ Asignación (=) ◦ Asignación de aumentación (+=, -=, *=, /=) ◦ Comparación (==, <, >, >=, <=, =, <>, in, not in, is, is not) ◦ Expresioneslógicas (and, or, not) • Si se utilizan operadores con prioridad diferente se aconseja rodear con espacios a los operadores de menor prioridad • No utilizar espacios alrededor del igual (=) cuando es utilizado para indicar un argumento de una función o un parámetro con un valor por defecto
Comentarios	• Los comentarios deben ser oraciones completas • Si un comentario es una frase u oración su primera palabra debe comenzar con mayúscula a menos que sea un identificador que comience con minúscula • Si un comentario es corto el punto final puede omitirse • Los comentarios de una línea para aclaraciones del código aparecerán seguidos de los caracteres “//” en caso de código JavaScript mientras que en Python por el carácter “#” y deben ubicarse en la misma línea que se desea comentar

Continúa en la siguiente página

Tipo de estándar	Descripción
Convenciones de nombramiento	• Nunca se deben utilizar como simples caracteres para nombres de variables los caracteres ele minúscula “l”, o mayúscula “O”, ele mayúscula “L” ya que en algunas fuentes son indistinguibles de los números uno (1) y cero (0) • Los módulos deben tener un nombre corto y en minúscula. • Los nombres de clases deben utilizar la convención “CapWords” (palabras que comienzan con mayúsculas) • Los nombres de las funciones deben estar escrito en minúscula separando las palabras con un guión bajo “_” • Las constantes deben quedar escritas con letras mayúsculas separando las palabras por un guión bajo (_)

A continuación, se muestra un fragmento de código donde se evidencia el uso de la codificación.

Código fuente 3.1. Ejemplo de código en Python

```
1 class TelgramRequestHandler(object):
2     def handle(self):
3         addr = self.client_address[0]          # Client IP-adress
4         telgram = self.request.recv(1024)      # Recieve telgram
5         print "From: %s, Received: %s" % (addr, telgram)
6         return
```

3.2. Diagrama de despliegue

Los diagramas de despliegue muestran cómo los componentes de software se despliegan físicamente en los procesadores; es decir, el diagrama de despliegue muestra el hardware y el software en el sistema, así como el middleware usado para conectar los diferentes componentes en el sistema. En esencia, los diagramas de despliegue se pueden considerar como una forma de definir y documentar el entorno objetivo (Sommerville, 2011).

A continuación, la figura 3.1 muestra el diagrama correspondiente al sistema propuesto.

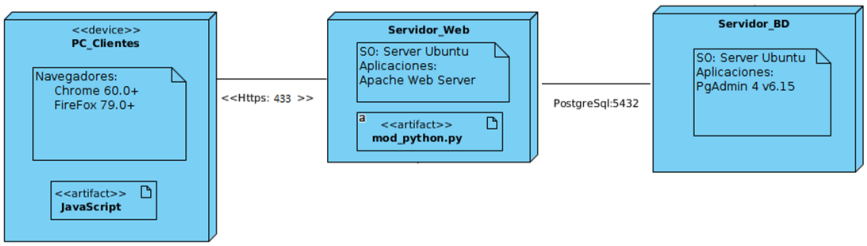


Figura 3.1. Representación del diagrama de despliegue.

PC_Clientes: Se refiere a el conjunto de todos los clientes que consumirán el software desde sus computadoras.
Sevidor_Web: Elemento de cómputo, dedicado al procesamiento y la lógica de la aplicación web, alimentada por la aplicación servidora.
Sevidor_BD: Elemento de cómputo, dedicado a almacenar y proveer datos necesarios para el funcionamiento de la aplicación web.

3.3. Pruebas de software

Las pruebas demuestran que un programa hace lo que se intenta que haga, así como descubrir defectos en el programa antes de usarlo. Hay que verificar los resultados de la prueba que se opera para buscar defectos, fallas o información de atributos no funcionales del programa (Sommerville, 2011).
A fin de encontrar los defectos del sistema y garantizar un nivel aceptable de calidad y confianza, se realizan pruebas de software, técnicas para el diseño de las pruebas, tanto manuales como estáticas, haciendo uso de las principales técnicas existentes, y aprovechando los módulos unitext.py y TestCase.py, ofrecidos por python y django respectivamente, que facilitan las pruebas automatizadas, apoyándose en la estrategia de pruebas recomendada por el framework.

3.3.1. Estrategia de pruebas

A continuación, se muestra la estrategia de prueba a seguir para validar la propuesta de solución.

Tabla 3.2. Estrategia de pruebas. (Fuente: Elaboración propia).

Pruebas	**Técnica para el diseño**	**Herramienta**	**Alcance**
Funcional	Caja negra con particiones equivalentes	Selenium IDE	Se prueba el funcionamiento del 100 % de los requisitos funcionales.
Unitaria	Caja blanca con la técnica del camino básico	Módulo TestCase de django para la realización de pruebas automatizadas	Se automatizarán pruebas para las unidades de código separadas por módulos.
Rendimiento	Pruebas de carga y estrés	Locust	Se aplican sobre un entorno de pruebas con prestaciones menores que el de despliegue, debido a falta de recursos. Se prueba la aplicación con 100 usuarios concurrentes buscando tiempos de respuesta menores a 2 segundos.

Continúa en la siguiente página

Pruebas	**Método**	**Herramienta**	**Alcance**
Seguridad		Acunetix web vulnerability scanner 9.5	Se aplica para detectar vulnerabilidades: • Inyección SQL • Programación Cross-Site (o XSS) • Ataques de fuerza bruta a las credenciales • Redirecciones y reenvíos no validados
Aceptación	Cuestionarios	Técnica de Iadov	Se le aplica la técnica con un cuestionario a un grupo de especialistas y trabajadores con años de experiencia en el proceso de promoción de la cultura. El éxito de la prueba se medirá: • Insatisfacción: desde (-1) hasta (-0,5) • Contradictorio: desde (-0.49) hasta (+0,49) • Satisfacción: desde (+0,5) hasta (1)
Ejecutado por:	Airely Sanchez Miralles		

3.3.2. Pruebas unitarias

Se aplican a un componente del software. Se puede considerar como componente, a una función, una clase, una librería, etc. Estas pruebas las ejecuta el desarrollador, cada vez que va probando fragmentos de código o scripts para ver si todo funciona como se desea. El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las partes individuales son correctas. Proporcionan un contrato escrito, que el fragmento de código debe satisfacer. El método utilizado para realizar este tipo de prueba se denomina caja blanca (Sommerville, 2011).

Método de caja blanca

Las pruebas de caja blanca intentan garantizar que:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo
- Se utilizan las decisiones en su parte verdadera y en su parte falsa
- Se ejecuten todos los bucles en sus límites
- Se utilizan todas las estructuras de datos internas.

Para la realización de las pruebas unitarias, se le aplica la técnica de prueba del camino básico a las unidades código que responden a funcionalidades críticas del software, lo cual permite generar el grafo de flujo, calcular la Complejidad Ciclomática (CC) para determinar los caminos linealmente independientes y el

número mínimo de escenarios de los casos de prueba para forzar la ejecución de cada camino del conjunto básico.

Luego en apoyo a las pruebas se utiliza el módulo TestCase que ofrece el framework, django, para la automatización de las pruebas unitarias. Con él se prueba cada módulo desarrollado, y a partir de la aplicación de la técnica de camino básico, en aquellas funcionalidades críticas, se pueden automatizar pruebas para cada uno de los escenarios o caminos posibles, garantizando probar todo el código en cuestión.

Entre los elementos de código que fueron probadas se encuentra el referente al método post de la clase Formulario_Delegación, que se encarga del controlar la lógica de añadir los Delegaciones Deportivas a desarrollarse dentro del Movimiento Deportivo en la Universidad.

Tabla 3.3. Cálculo de la complejidad ciclomática del método post de la clase Agregar_Evento

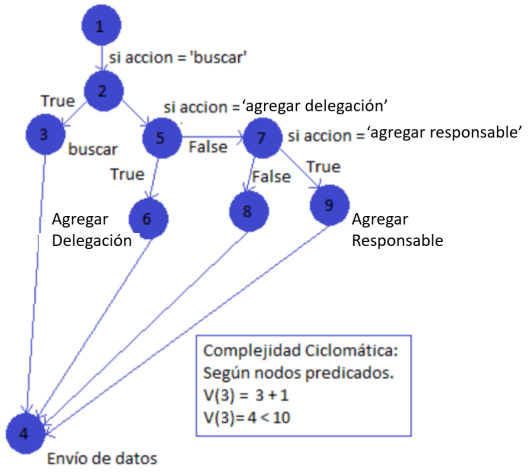
Método

```
class Formulario_Delegacion(View):
    form_class = DelegacionDeportivaForm
    template_name = 'FormulariosHTML/crear_delegacion.html'
    success = ''

    def get(self, request):
        # Generar formulario
        form = self.form_class()
        return render(request, self.template_name, {"form": form})

    def post(self, request):
        if request.method == "POST":
            formulario = DelegacionDeportivaForm(request.POST)
            if formulario.is_valid():
                formulario.save()
                return redirect("delegaciones")
            else:
                form = self.form_class()
                return render(request, self.template_name, {"form": form})
```

Grafo resultante:



Complejidad Ciclomática:	$V(G) = A - N + 2$	$V(G) = P + 1$
$V(G) = \#\ de\ regiones$	$V(G) = 11 - 9 + 2$	$V(G) = 3 + 1$
$V(G) = 4$	$V(G) = 4$	$V(G) = 4$

Luego de la determinación de los nodos y flujos de control del código se obtuvo el grafo de flujo y se calculó la complejidad ciclomática del algoritmo.
Como resultado se obtiene que la CC es igual a 4, lo que significa que existen cuatro posibles caminos linealmente independientes y hay que diseñar un mínimo de cuatro casos de prueba para el algoritmo. La tabla 3.4 muestra los caminos existentes.

Tabla 3.4. Caminos del grafo de flujo (Fuente: Elaboración propia).

No.	Camino
1	1,2,3,4
2	1,2,5,6,4
3	1,2,5,7,8,4
4	1,2,5,7,9,4

Los casos de prueba para las pruebas de caja blanca por la técnica de camino básico se ejecutan por cada camino independiente que se determine en un algoritmo específico. A continuación, se muestra el caso de prueba para el camino básico independiente 1 del algoritmo.

Tabla 3.5. Caso de Prueba para el camino básico 1 (Fuente: Elaboración propia).

Proceso:
Agregar Delegación.
Casos de prueba:
Buscar Delegación. Escenario 1.2.
Camino independiente:
1,2,5,6,4
Entradas:
• Deportista: id
Resultados esperados:
• El sistema añade una Delegación Deportiva al sistema. • En la UI se muestra la Delegación añadida en la lista de delegaciones.
Condiciones de ejecución:

- Usuario autenticado.
- Usuario con rol de solicitante o responsable.

Con la realización de los casos de prueba diseñados se prueba la ejecución de cada sentencia del código al menos una vez, teniendo en cuenta todas las condiciones lógicas en sus variantes verdaderas y falsas. La obtención de la CC de valor 4 del método post ejemplificado, permitió determinar que existen 4 caminos linealmente independientes, suficientes para probar el código al menos una vez.
Los resultados del método de caja blanca fueron satisfactorios. Se automatizaron un total de 27 casos de prueba con el uso de la biblioteca TestCase, de los cuales a 5 se le aplicó la técnica del camino básico, permitiendo que su automatización garantice probar todos los caminos con un mínimo de escenarios diseñados, y obteniendo 0 defectos.

```
----------------------------------------------------------------------
Ran 50 test in 9.209s

OK
Destroying test database for alias 'default'...
```

Figura 3.2. Resultado de las pruebas unitarias.

3.3.3. Pruebas funcionales

Este tipo de prueba se realiza sobre el sistema funcionando, comprobando que cumpla con la especificación. Para estas pruebas, se utilizan las especificaciones de casos de prueba: se intenta demostrar que el sistema implementó adecuadamente sus requisitos. (Sommerville, 2011).

Método de caja negra

Las pruebas de caja negra, también llamadas pruebas de comportamiento, se enfocan en los requerimientos funcionales del software. Las técnicas de prueba de caja negra permiten derivar conjuntos de condiciones de entrada que revisarán los requerimientos funcionales para un programa (Pressman, 2005). El método de caja negra presenta varias técnicas de prueba como son: partición de equivalencia, análisis de valores límites y grafos de causa-efecto entre otros.
En la presente investigación se utiliza específicamente dentro del método de caja negra la técnica de partición de equivalencia generando los casos de pruebas de dicha técnica sobre las diferentes interfaces que responden a los requisitos funcionales. Para la aplicación de pruebas de regresión sobre los casos de prueba definidos se utiliza la herramienta Selenium IDE, que permite grabar todas las interacciones de un usuario con el navegador y posibilita ejecutar de forma automática las mismas, reduciendo el tiempo y los costos de las pruebas funcionales.

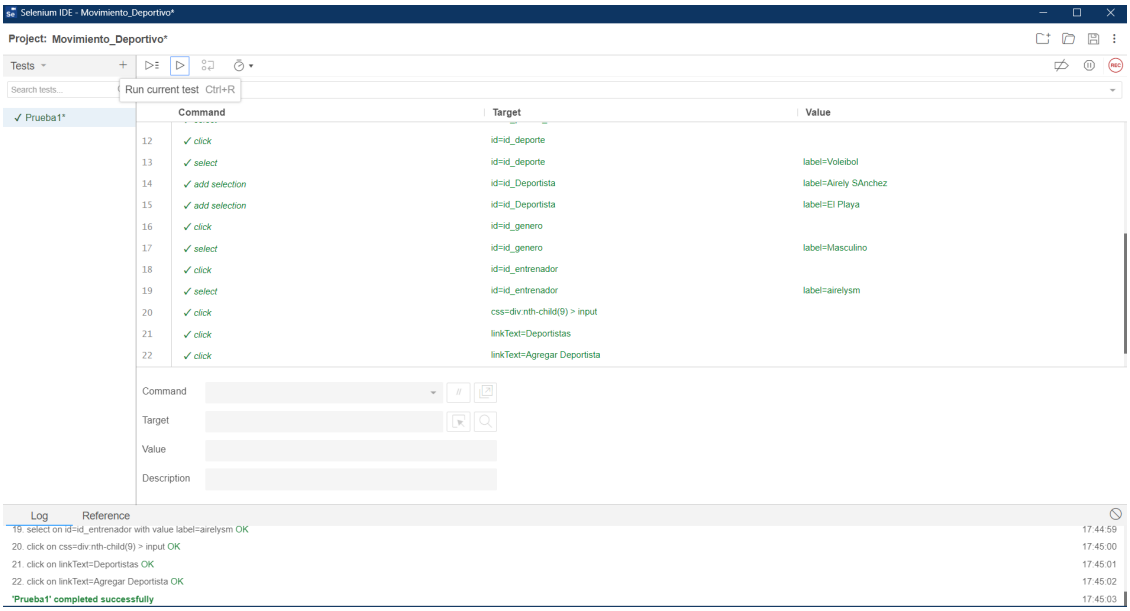


Figura 3.3. Representación del resultado la ejecución de una prueba usando Selenium IDE, Crear Delegación Deportiva.

A continuación, la tabla 3.6 muestra el diseño de caso de pruebas del requisito “Añadir Integrante a una Delegación Deportiva” donde se analizarán las variables y condiciones que puedan determinar la respuesta del sistema.

Tabla 3.6. Caso de prueba “Añadir Integrante” (Fuente: Elaboración Propia).

Escenario	Descripción	Variables		Respuesta esperada	Respuesta
		Integrante	Estado		
EC 1.1. Añadir Integrante correctamente	El Usuario debe seleccionar al menos un integrante para luego dar clic a en botón “Guardar”	V Airely Sanchez Miralles	V Estudiante	Se envían los datos al servidor	El sistema añade al integrante a la Delegación Deportiva
EC 1.2. Añadir Integrante incorrectamente		I No se seleccionan integrantes	V Se deja el campo en blanco	Se presiona el botón Guardar y el sistema no envía los datos	Al no seleccionar algunos campos obligatorios el sistema no envía los datos al servidor

Tabla 3.7. Variables de caso de prueba "Añadir Integrantes a una Delegación Deportiva"(Fuente: Elaboración Propia).

No.	Variable	Valor Nulo	Descripción
1	Integrantes	No	Es un campo checkbox que permite al usuario elegir de un listado de personas que cumplan ciertas características las más adecuadas para participar en la Delegación Deportiva que está definiendo o modificando
2	Género	Si	Es un campo que permite al usuario seleccionar el género de los integrantes dentro de la Delegación Deportiva.

Las pruebas de caja negra se aplican con el objetivo de evaluar las interfaces de comunicación con el usuario, las que demostraron coherencia y funcionalidad, así como probar todas aquellas funcionalidades directamente relacionadas con los requisitos funcionales del sistema. La técnica de partición de equivalencia es aplicada para evaluar los diferentes escenarios que pueden tener lugar ante la ejecución de una acción. Como resultado de la aplicación de estas pruebas se ejecutan las posibles variantes que posee una interfaz de comunicación con el usuario, resolviendo los defectos arrojados y perfeccionando lo obtenido.
Durante la realización de las pruebas se detectan un conjunto de defectos relacionados con errores de validación y funcionalidad. Los resultados se muestran en la figura 3.4, donde se evidencia la cantidad de casos de prueba ejecutados y los casos de prueba que fallaron. Se realizaron cuatro iteraciones, durante la primera iteración se analizaron 27 casos de prueba, de los cuales 19 fallaron. En la segunda y tercera iteración a través de las pruebas de regresión, con el uso del software Selenium IDE, se verifica que las no conformidades anteriores estuviesen solucionadas, y de estas pruebas se obtienen 13 y 5 nuevos defectos respectivamente, quedando resuelta en la cuarta iteración y cumpliéndose correctamente los requisitos funcionales.

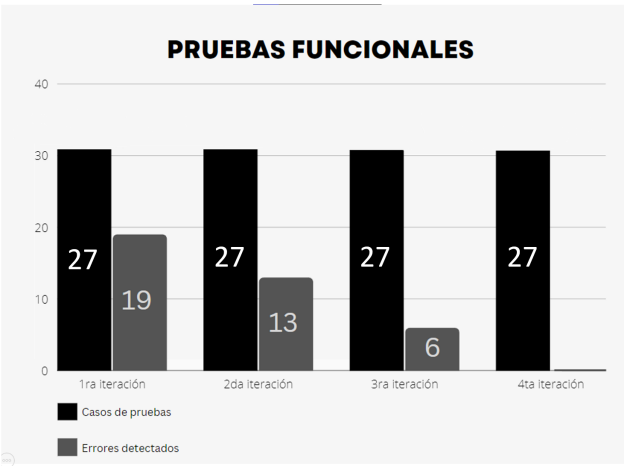


Figura 3.4. Representación del resultado de las pruebas.

3.3.4. Pruebas de Integración

Las pruebas de integración son un tipo de pruebas de software que se realizan para comprobar que los diferentes componentes o módulos de una aplicación funcionan correctamente cuando se combinan entre sí. Estas pruebas se hacen después de las pruebas unitarias, que verifican el funcionamiento de cada componente por separado. El objetivo de las pruebas de integración es detectar y corregir los posibles defectos que puedan surgir al integrar los componentes, como problemas de comunicación, de compatibilidad, de rendimiento o de funcionalidad. Las pruebas de integración pueden seguir diferentes estrategias, como la integración incremental, la integración en paralelo, la integración ascendente, la integración descendente o la integración en sándwich. Estas estrategias definen el orden y la forma en que se integran los componentes. Las pruebas de integración pueden ser manuales o automatizadas, dependiendo de la complejidad del sistema y de los recursos disponibles (Ramírez Quesada, 2023).

En el proyecto del Sistema Gerencial de Extensión Universitaria se ubican los componentes en los directorios correspondientes, garantizando la correcta integración del mismo.

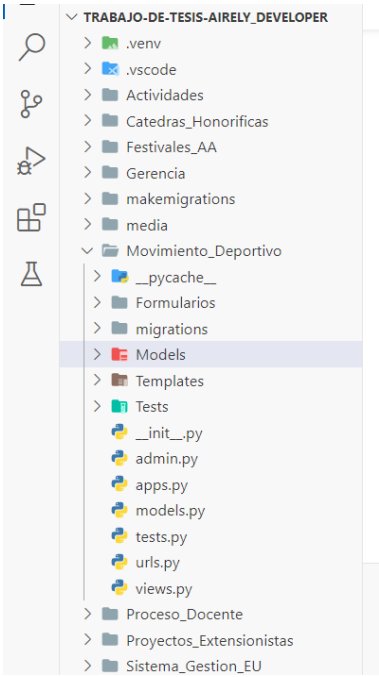


Figura 3.5. Representación de la integración del directorio del Módulo en cuestión al proyecto.

El directorio más importante es Gerencia, aquí es donde se cargan todos los estilos además de la gestión de los componentes del proyecto. Algunos de los directorios más importantes también son:

- **Media:** Gestiona la parte de los datos guardados correspondiente a la multimedia que se solicita a través de los diferentes módulos.

- **Sistema_Gestion_EU:** Administra las URLS de cada uno de los módulos del proyecto así como la configuración y gestión de las conexiones con la Base de Datos.

La estructura del proyecto permite definir como se integran y se relacionan los componentes con los diferentes directorios. Mediante la aplicación de la técnica de integración primero en anchura se establece un diagrama de tres niveles con los principales componentes que se integran (ver figura 3.6).

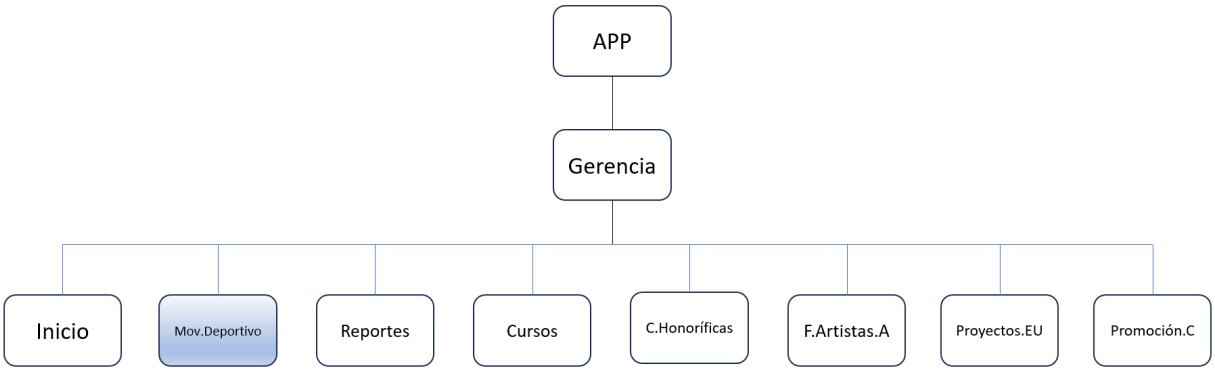


Figura 3.6. Integración Descendente.

Los componentes Inicio, Mov.Deportivo, Reportes, Cursos, C.Honoríficas, F.Artistas.A, Proyectos.EU y Promoción.C se integran primero al módulo Gerencia y este a su vez al nivel de control APP. El proceso de integración se realiza en una serie de cinco pasos:

1. El módulo de control principal se utiliza como controlador de prueba y los restantes se sustituyen con todos los componentes directamente subordinados al módulo de control principal.
2. Los demás módulos se sustituyen uno a la vez con componentes reales atendiendo al enfoque primero en anchura.
3. Las pruebas se llevan a cabo conforme se integra cada componente.
4. Al completar cada conjunto de pruebas, otro módulo se sustituye con el componente real.
5. Las pruebas de regresión pueden realizarse para asegurar que no se introdujeron nuevos defectos.

3.3.5. Prueba de seguridad

Las pruebas de seguridad se diseñan para sondear las vulnerabilidades del entorno lado cliente, las comunicaciones de red que ocurren conforme los datos pasan de cliente a servidor y viceversa, y el entorno del

lado servidor. Cada uno de estos dominios puede atacarse, y es tarea del examinador de seguridad descubrir las debilidades que puedan explotar quienes tengan intención de hacerlo (Pressman, 2005).
Las pruebas de seguridad se aplican con ayuda de la herramienta Acunetix Web Vulnerability Scanner 11.0.1 que establece alertas de tipo: alta, media, baja e informacional, realizándose en dos iteraciones durante el desarrollo de la propuesta solución.
En una primera iteración se obtiene un total de 22 alertas de seguridad, de las cuales 4 clasifican de nivel medio, 1 de nivel bajo y 17 informativas. De las de nivel medio, destacó el uso de protocolo no seguro para el envío de datos, así como los mensajes de error que se muestra en el modo DEBUG de Django para el desarrollo y se detectaron problemas para la protección de contra ataques de fuerza bruta en el formulario de autenticación.
La de nivel bajo, consistía en vistas del sitio que se podían acceder directamente sin pasar la autenticación y el sistema de roles establecido. De carácter informativo fueron detectadas posibles cuentas de usuario en ficheros, así como presencia de directorios desprotegidos y la existencia de etiquetas iframe de HTML5.

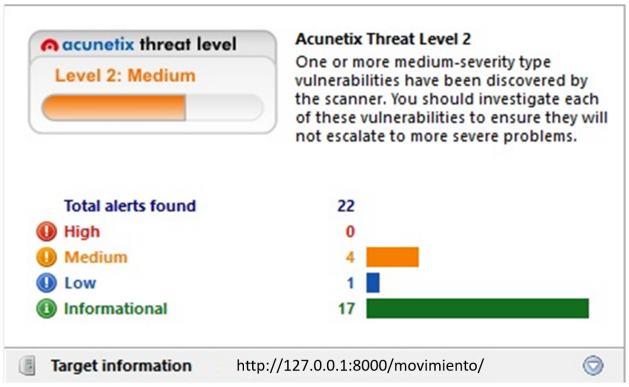


Figura 3.7. Prueba de seguridad 1ra iteración.

Después de aplicar refactorización del código y realizar las validaciones correspondientes, se aplicó la segunda iteración en búsqueda de vulnerabilidades al sistema, arrojando como resultado que todas las que se habían detectado en la primera iteración, habían sido corregidas.

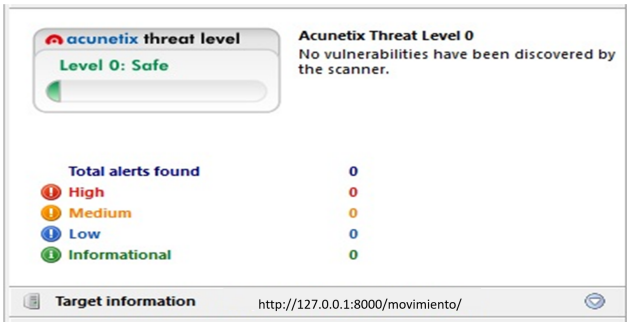
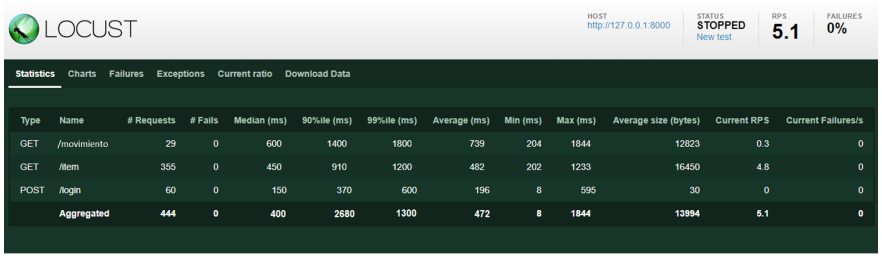


Figura 3.8. Prueba de seguridad 2da iteración.

3.3.6. Prueba de rendimiento

Las pruebas de rendimiento deben diseñarse para garantizar que el sistema procese su carga pretendida. Esto implica efectuar una serie de pruebas donde se aumenta la carga, hasta que el rendimiento del sistema se vuelve inaceptable. Las pruebas de rendimiento se preocupan tanto por demostrar que el sistema cumple con sus requerimientos, como por descubrir problemas y defectos en el sistema (Sommerville, 2011).
Para la realización de las pruebas de rendimiento del sistema se utiliza el paquete de python locust destinado para la ejecución de estas pruebas mediante código python. El cuál permite probar la aplicación simulando un entorno similar al de producción o despliegue de la aplicación, donde actúan de forma concurrente 100 usuarios, realizando alrededor de 5 peticiones por segundo, obteniendo un tiempo de respuesta máximo menor que dos segundos, cumpliendo así con lo pactado con el cliente en los requisitos no funcionales del sistema.



Type	Name	# Requests	# Fails	Median (ms)	90%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/movimiento	29	0	600	1400	1800	739	204	1844	12823	0.3	0
GET	/item	355	0	450	910	1200	482	202	1233	16450	4.8	0
POST	/login	60	0	150	370	600	196	8	595	30	0	0
	Aggregated	**444**	**0**	**400**	**2680**	**1300**	**472**	**8**	**1844**	**13994**	**5.1**	**0**

Figura 3.9. Prueba de rendimiento con 100 usuarios concurrentes.

A fin de comprobar el comportamiento del sistema sobre una carga superior, hasta llevarlo al límite, se le aplicó la prueba de estrés, con la presencia de 200 usuarios concurrentes, conectándose de uno en uno por cada segundo, y se obtuvo como resultado que, el sistema superó la barrera de 3 segundos de respuesta a los 160 usuarios concurrentes y comenzó a presentar fallos en las respuestas a los clientes a partir de los 190 usuarios, llegando a un 2 **por ciento** de fallos con respecto al total de peticiones con los 200 usuarios en el sistema.

Type	Name	# Requests	# Fails	Median (ms)	90%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/movimiento	46	2	13000	31000	40000	14427	298	40384	21708	0.3	0.2
GET	/item	589	12	11000	20000	28000	10549	191	35906	20271	4.3	0.9
POST	/login	133	0	280	620	1200	319	7	1347	30	1	0
	Aggregated	**768**	**14**	**8000**	**20000**	**30000**	**9010**	**7**	**40384**	**16852**	**5.6**	**1.1**

Figura 3.10. Prueba de rendimiento con 200 usuarios concurrentes.

Con la aplicación de las pruebas de rendimiento se arroja que el sistema responde como se esperaba bajo una carga normal de 100 usuarios concurrentes en condiciones similares a las de producción o despliegue

del sistema, respondiendo a los usuarios en no más de 2 segundos. Además la realización de una prueba de estrés complementaria, permitió saber el momento justo en el que el sistema empieza a fallar y a obtener tiempos de respuesta inaceptables para una correcta experiencia de usuario, superiores a los 3 segundos.

3.3.7. Pruebas de aceptación

Para la revisión final de las especificaciones del diseño y de la implementación se realizaron las pruebas de aceptación con el objetivo de determinar diferentes tipos de defectos y resolverlos (Pressman, 2005), para ello se emplean las pruebas Alfa y Beta.

- **Pruebas Alfa:** Se desarrollan en conjunto, el desarrollador y los usuarios finales. Con el objetivo de registrar los errores y problemas del uso del software.
- **Pruebas Beta:** El sistema fue desplegado en un servidor del centro cultural, con 10 clientes conectados en red local, para la realización de las pruebas. A diferencia de la prueba alfa, la prueba beta es una aplicación del software en su ambiente final. El objetivo es registrar todos los problemas que se encuentran durante la prueba.

Para la aplicación de las pruebas de aceptación se seleccionó un grupo de profesores y especialistas que han impartido durante varios años diversos cursos en DEU de la UCI. A continuación, se relacionan los involucrados en las pruebas:

- MsC. Julio César Espronceda Pérez (Director de Extensión Universitaria)
- Ing. Alvaro Alejnadro Acosta Ruiz (Especialista B en Ciencias Informáticas)
- MsC. Mirta Beltrandez Sardiñas (Especialista B en Ciencias Informáticas)
- Lic. Yanaida Delgado Gonzáles (Especialista Superior)
- Lic. Berta Zelaida López Pérez (Especialista Superior)
- Lic. Madelaine Bárbara Labadié Martinez (Especialista General)
- MsC. Antonio Gutiérrez Laborit (Vicerrector de Extensión)
- Lic. Aloy Machado Sánchez (Director de Deportes)
- Lic. Jorge Rodríguez Navarro (Especialista Superior)
- Ing. Allan Pierra Fuentes (Director de Soporte Tecnológico)

Como resultado después de haber aplicado las pruebas Alfa y Beta se identificaron nuevas no conformidades, las cuales fueron resueltas. Las pruebas Alfas aplicadas en un ambiente controlado, donde se probó por parte del equipo de desarrollo la funcionalidad del sistema, arrojaron defectos en el cumplimiento de los requisitos no funcionales. Mientras que las pruebas Beta aplicadas en el ambiente final donde será desplegada la solución arrojaron defectos en el proceso de búsqueda de un Evento Deportivo y el nivel de aprobación de estas.

3.4. Satisfacción de los usuarios

Con el objetivo de evaluar el sistema implementado se utiliza la técnica de Iadov, esta técnica evalúa el nivel de satisfacción del usuario, permitiendo conocer si la solución propuesta cumple con las expectativas esperadas. Esta técnica constituye una vía indirecta para el estudio de la satisfacción, al ser los criterios que se utilizan se fundamentan en las relaciones que se establecen entre tres preguntas cerradas (preguntas 1, 2 y 3) que se intercalan dentro de un cuestionario (Ver Anexo .1) (Gallegos; Valenzuela; López; Richart y Alonso, 2006). Estas tres preguntas se relacionan a través de lo que se denomina el "Cuadro Lógico de Iadov", el cual se muestra a continuación en la tabla 3.8.

Tabla 3.8. Cuadro Lógico de Iadov (Fuente: Elaboración propia).

	1- ¿Consideras adecuado el uso de las TIC en el proceso de informatización del Movimiento Deportivo?								
	NO			NO SÉ			SÍ		
3- ¿Satisfacen sus necesidades como organizador deportivo, el sistema propuesto?	2- ¿Utilizaría usted la aplicación propuesta a la hora de gestionar el proceso del movimiento deportivo en la UCI?								
	SÍ	NO SÉ	NO	SÍ	NO SÉ	NO	SÍ	NO SÉ	NO
Me satisface mucho	1	2	6	2	2	6	6	6	6
No me satisface tanto	2	3	2	3	3	3	6	3	6
Me da lo mismo.	3	3	3	3	3	3	3	4	3
Me disgusta más de lo que me satisface.	6	3	6	3	4	4	3	4	4
No me satisface nada.	6	6	6	6	4	4	6	4	5
No sé qué decir.	2	3	6	3	3	3	6	3	4

El número resultante de la interrelación de las tres preguntas indica la posición de cada sujeto en la escala de satisfacción.
Escala de satisfacción:

- Clara satisfacción (1).
- Más satisfecho que insatisfecho (2).
- No definida (3).
- Más insatisfecho que satisfecho (4).
- Clara insatisfacción (5).
- Contradictoria (6).

Para medir el grado de satisfacción se toma una muestra de 10 personas. Las personas que son seleccionadas tienen más de 5 años de experiencia en el trabajo de la extensión universitaria. De ellas 3 son Máster en Ciencias, 3 son Ingenieros y el resto son Licenciados.

La técnica de Iadov permite conocer el Índice de satisfacción grupal (ISG), para lo cual se trabaja con los diferentes niveles de satisfacción que se expresan en una escala numérica que oscila entre +1 y -1 de la siguiente forma:

Índice	**Escala**
+1	Máximo de satisfacción.
0.5	Más satisfecho que insatisfecho.
0	No definido y contradictorio.
-0.5	Más insatisfecho que satisfecho.
-1	Máxima insatisfacción.

La satisfacción grupal se calcula utilizando la siguiente fórmula:

$$ISG = \frac{A\,(+1) + B\,(+0,5) + C\,(0) + D\,(-0,5) + E\,(-1)}{N}$$

Donde:

- A, B, C, D, E representan el número de sujetos con índice individual 1, 2 , 3 o 6, 4, 5 respectivamente.
- N representa el número total de sujetos del grupo.

Esto permite reconocer las categorías grupales:

- Insatisfacción: desde (-1) hasta (-0,5)
- Contradictorio: desde (-0.49) hasta (+0,49)
- Satisfacción: desde (+0,5) hasta (1).

Luego de haber aplicado la técnica, calculando el ISG se obtiene como resultado 0.8 lo que significa una clara satisfacción con el uso del sistema de gestión del Movimiento Deportivo.

$$ISG = \frac{7\,(+1) + 2\,(+0,5) + 1\,(0) + 0\,(-0,5) + 0\,(-1)}{10} = 0.8$$

A partir de la realización de las pruebas y la aplicación de la técnica Iadov, se determinó que el sistema informático cumplió con los indicadores establecidos de funcionalidad, obteniendo el máximo calificativo en todos los aspectos, además se validó que el proceso de gestión de la información se garantiza en sus tres dimensiones, de organización de la información, centralización, mediante un diseño de una base de datos robusta y normalizada, sin duplicación de la información, así como la disponibilidad de la misma, con los tiempos de respuesta esperados, respaldados por las pruebas de rendimiento.

3.5. Consideraciones parciales

En este capítulo se abordaron los elementos del Sistema para la gestión del Movimiento Deportivo, así como las pruebas realizadas al mismo y los resultados obtenidos; lo que permite concluir:

- El estándar de codificación y diagrama de despliegue de la propuesta de solución facilitó obtener una aplicación funcional capaz de contribuir con la gestión del Movimiento Deportivo dentro de la UCI
- La validación de la propuesta de solución, mediante una estrategia de pruebas de software facilitó corroborar la calidad de la misma, estableciendo así el cumplimiento de los requisitos definidos por el cliente.
- La aplicación de la técnica de Iadov aplicada a 10 usuarios arrojó como resultado un alto índice de satisfacción de la solución propuesta por parte de un grupo de profesores y especialistas de la DEU en la UCI, lo cual se traduce en una clara satisfacción con el uso del sistema informático propuesto.

Conclusiones

El presente Trabajo de Diploma concluye con el desarrollo de un Módulo para la Gestión del Proceso del Movimiento Deportivo para el Sistema Gerencial de Extensión Universitaria.

- El estudio asociado al proceso de gestión del Movimiento Deportivo, permitió definir que la propuesta de solución deberá ser de tipo web.
- El análisis de los sistemas homólogos permitió identificar las principales funcionalidades y características que debe tener el sistema a desarrollar.
- Las herramientas y tecnologías a utilizar asegura el ambiente de desarrollo necesario para la implementación de la propuesta de solución, destacando como lenguajes de programación HTML5, CSS3, Python, JavaScript y como framework de desarrollo Django.
- La descripción de la propuesta de solución mediante la investigación y análisis de sus particularidades fundamentales, posibilitó identificar el carácter funcional del sistema propuesto, acotado por las necesidades del cliente para colaborar a la gestión del Movimiento Deportivo de la Universidad de las Ciencias Informáticas.
- La validación de la propuesta de solución, mediante una estrategia de pruebas de software facilitó corroborar la calidad de la misma, estableciendo así el cumplimiento de los requisitos definidos por el cliente.

Recomendaciones

Para dar continuidad a la presente investigación se recomienda:

- Generalizar el uso del Sistema Gerencial de Extensión Universitaria en todas las IES.

Acrónimos

AJAX Xivas Ascícronos de JavaScript. 14
API Progrma de Interfáz de Aplicación. 14
AUP Proceso Unificado Ágil. 11, 15
AUP-UCI Proceso Unificado Ágil versión UCI. 20

CASE Ingeniería de Software Asistida por Computadora. 11
CC Complejidad Ciclomática. 35–37
CD Cursos Diurnos. 10
CES Centros de Educación Superior. 10
CPE Curso por Encuentros. 10
CSS Hoja de estilo en Cascada. 12, 13, 23
CSS3 Hoja de estilo en Cascada Versión 3. 19

DEU Dirección de Extensión Universitaria. 6, 15, 16, 43, 46

EU Extensión Universitaria. 5, 6, 9, 16

FEU Federación Estudiantil Universitaria. 2, 7

GI Gestión de la Información. 7, 10
GOF Pandilla de cuatro. 26
GRASP Patrones Generales de Software de Asignación de Responsabilidad. 25

HTML Lenguaje de marcado de hipertexto. 12–14, 24
HTML5 Leguaje de marcado de hipertexto Versión 5. 20, 40
HU Historias de Usuario. 20, 28

IDE Entorno de Desarrollo Integrado. 12
IES Instituciones de Educación Superior. 4, 55, 56
INDER Instituto Nacional de Deporte, Educación Física y Recreación. 7
ISA Instituto Superior de Arte. 9
ISDI Instituto Superior de Diseño Industrial. 10
ISG Índice de satisfacción grupal. 44, 45

ISRI Instituto Superior de Relaciones Internacionales. 10

JSON Notación de objetos JavaScript. 12

. 6–8, 11, 15
MES Ministerio de Educación Superior. 2, 6, 10
MTV Modelo Vista Plantilla. 22, 23
MVC Modelo Vista Controlador. 22, 23

ORM Mapeador de Objeto Relacional. 25

PHP Preprocesador de Hipertexto. 12
PNEU Programa Nacional de Extensión Universitaria. 6
POO Programación Orientada a Objeto. 26

SGEU Sistema Gerencial de Extensión Universitaria. 15
SIGENU Sistema de Gestión Universitaria. 10
SIGUA Sistema informático de gestión universitaria. 9
SNIES Sistema Nacional de Información de la Educación Superior. 8
SPADIES Sistema para la Prevención de la Deserción en la Educación Superior. 8, 9
SQL Lenguaje de Consultas Estructurado. 14

TIC Tecnologías de la Información y las Comunicaciones. 1, 44, 53

UCI Universidad de las Ciencias Informáticas. 2–7, 11, 14–16, 28–47, 53
UI Interfaz de Usuario. 19, 21, 36, 55
UML Lenguaje de Modelado Unificado. 11, 20, 23

Referencias bibliográficas

BALOGUN, MO, 2022. Comparative Analysis of Complexity of C++ and Python Programming Languages. *Asian J. Soc. Sci. Manag. Technol*. Vol. 4, págs. 1-12 (vid. pág. 12).

BARZAGA-SABLÓN, Oscar Santiago; PINCAY, Hugo Jesús Juan Vélez; NEVÁREZ-BARBERÁN, José Víctor y COBEÑA, María Verónica Arroyo, 2019. Gestión de la información y toma de decisiones en organizaciones educativas. *Revista de ciencias sociales*. Vol. 25, n.º 2, págs. 120-130 (vid. pág. 7).

CAMPMANY CEJAS, Alejandro, 2022. *Sistema informático para la gestión del Movimiento de Artistas Aficionados en la Universidad de las Ciencias Informáticas*. B.S. thesis. Universidad de las Ciencias Informáticas. Facultad 1 (vid. pág. 2).

CASTELLÓN, Juan Antonio Hernández y DÍAZ, Winston Joseph Zamora, 2020. Sistemas de gestión de la calidad: Una mejora en la calidad de las Instituciones de Educación Superior en Nicaragua. *Revista Electrónica de Conocimientos, Saberes y Prácticas*. Vol. 3, n.º 2, págs. 87-97 (vid. pág. 1).

EDUCACIÓN NACIONAL DE COLOMBIA, Ministerio de. Sistema Nacional de Información de la Educación Superior (SNEIS)[Internet (vid. pág. 8).

ESPRONCEDA PÉREZ, Julio César; PÉREZ FUENTES, Armando y REMEDIO FROMETA, Sulieka, 2021. Estrategia para la gestión del talento deportivo en la Universidad de las Ciencias Informáticas (vid. págs. 1, 7).

FARANELLO, Scott, 2012. *Balsamiq wireframes quickstart guide*. Packt Publishing (vid. pág. 11).

GALLEGOS, Antonio Granero; VALENZUELA, Alfonso Valero; LÓPEZ, Manuel Gómez; RICHART, Cecilia Barrachina y ALONSO, Saula Jurado, 2006. Las clases de Educación Física y el deporte extraescolar entre el alumnado almeriense de primaria. Una aplicación práctica mediante la técnica de Iadov. *Lecturas: Educación física y deportes*. N.º 98, págs. 8 (vid. pág. 44).

GAMMA, Erich; HELM, Richard; JOHNSON, Ralph; JOHNSON, Ralph E; VLISSIDES, John et al., 1995. *Design patterns: elements of reusable object-oriented software*. Pearson Deutschland GmbH (vid. pág. 23).

GÓMEZ HERNÁNDEZ, Celia Rosa; CÉSPEDES LARA, Leandro Lázaro; LLANO CASTRO, Eileén; CRUZ AMARÁN, Damaris y TERRERO GALANO, Roberkis, 2015. *Solución para la gestión de información de la Extensión Universitaria en el área de Cultura Física en la Universidad de las Ciencias Informáticas*. B.S. thesis (vid. pág. 10).

GONZÁLEZ APORTELA, Odette; BATISTA MAINEGRA, Amado y GONZÁLEZ FERNÁNDEZ-LARREA, Mercedes, 2020. Indicadores de calidad del proceso de extensión universitaria en la Universidad de La Habana. *Revista San Gregorio*. N.º 43, págs. 49-64 (vid. pág. 5).

GONZÁLEZ REYES, Alena y FEBLES ESTRADA, Ailyn, 2021. Principales áreas de impacto y resultados de la informatización de la sociedad cubana (vid. pág. 1).

HUNT, John, 2013. Gang of four design patterns. En: *Gang of four design patterns*. *Scala design patterns*. Springer, págs. 135-136 (vid. págs. 24, 25).

JACOB THORNTON, Mark Otto, 2022. *Get started with Bootstrap* [online] [visitado 2022-06-27]. Disponible desde: `https://getbootstrap.com/docs/5.2/getting-started/introduction/` (vid. pág. 13).

JAISWAL, S, 2016. *Javatpoint. com*. Javatpoint (vid. pág. 21).

JIMÉNEZ[1], Javier Guillot. SIGUA: SISTEMA INFORMÁTICO DE GESTIÓN UNIVERSITARIA (vid. pág. 8).

JOVENCLUB, Revista, 2020. *Gestion de información*. Url: `https://revista.jovenclub.cu/gestion-de-informacion/` (vid. pág. 7).

KAPLÚN, Gabriel, 2013. La integralidad como movimiento instituyente en la universidad. *Universidad de la República, Ciencias de la Comunicación*. Vol. 1, págs. 8. ISSN 2301-0118. (vid. pág. 5).

KOÇ, Hatice; ERDOĞAN, Ali Mert; BARJAKLY, Yousef y PEKER, Serhat, 2021. UML diagrams in software engineering research: a systematic literature review. *Multidisciplinary Digital Publishing Institute Proceedings*. Vol. 74, n.º 1, págs. 13 (vid. pág. 11).

LARMAN, Craig, 2003. UML y Patrones (vid. pág. 24).

LOZANO, MsC. Fernando Picayo, 2018. *Comunidad sigenu*. Url: `http://sigenu-comunidad.cujae.edu.cu:8081/sigenu-comunidad` (vid. pág. 9).

MALQUI, Diego David, 2018. Diseño y desarrollo Web accesible utilizando HTML5 y CSS3 con nivel de conformidad A, respecto a las pautas WCAG (vid. pág. 12).

MES, 2004. *Programa Nacional de Extension Universitaria* (vid. págs. 2, 6).

MES, 2022. *Ministerio | MES* [online] [visitado 2022-04-12]. Disponible desde: `https://www.mes.gob.cu/ministerio` (vid. págs. 2, 7).

MICROSOFT, 2022. *Get Started with Visual Studio Code* [online] [visitado 2022-06-08]. Disponible desde: `https://code.visualstudio.com/learn/overview` (vid. pág. 12).

MILLER, Curtis G, 2022. Introduction to Git (vid. pág. 14).

NISKA, Christoffer, 2014. *Extending Bootstrap*. Packt Publishing (vid. pág. 13).

PARADIGM, Visual, 2022. *About Visual Paradigm* [online] [visitado 2022-06-24]. Disponible desde: `https://www.visual-paradigm.com/aboutus/` (vid. pág. 11).

PÉREZ, Javier Eguíluz, 2019. *introduccion a JavaScript* (vid. pág. 12).

PÉREZ, Maidelyn Díaz, 2017. Sistemas de gestión de información y conocimiento en empresas cooperativas: sociedades colaborativas de conocimiento. *Cooperativismo y Desarrollo: COODES*. Vol. 5, n.º 2, págs. 221-232 (vid. pág. 8).

PILICITA GARRIDO, Anabel; BORJA LÓPEZ, Yolanda y GUTIÉRREZ CONSTANTE, Gonzalo, 2021. Rendimiento de MariaDB y PostgreSQL (vid. pág. 14).

PRESSMAN, Roger S, 2005. *Software engineering: a practitioner's approach*. Palgrave macmillan (vid. págs. 11, 23, 25, 36, 41, 43).

PUIG, Jordi Collell, 2013. CSS3 y Javascript avanzado. *Diponible en https://openlibra. com/es/book/download/css3-y-javascript-avanzado* (vid. pág. 13).

QUIÑONES LAFFITA, Arney; HERNÁNDEZ TORRES, Iosvani y CORDÓN GONZÁLEZ, Francisco, 2018. Uso de la tecnología en la gestión extensionista, efectividad de la página web extensión universitaria. *Revista Cubana de Ciencias Informáticas*. Vol. 12, págs. 176-189 (vid. pág. 9).

RAMÍREZ QUESADA, Úrbez Jesús, 2023. Diseño y desarrollo de soluciones DevOps para la automatización de procesos de desarrollo y despliegue de software en sistemas corporativos. (Vid. pág. 39).

RUBIO, Daniel, 2017. *Beginning Django*. Springer (vid. pág. 13).

SOMMERVILLE, Ian, 2011. *Ingeniería de software* [online] [visitado 2022-06-18]. ISBN 978-607-32-0604-4 978-607-32-0603-7. Disponible desde: `http://www.ingebook.com/ib/NPcd/IB_Escritorio_Visualizar?cod_primaria=1000193&libro=1518`. OCLC: 1257013649 (vid. págs. 20, 21, 31-33, 36, 42).

TABARES, Ricardo Botero, 2010. Patrones grasp y anti-patrones: un enfoque orientado a objetos desde logica de programacion. *Entre Ciencia e Ingeniería*. Vol. 4, n.º 8, págs. 161-173 (vid. págs. 23, 24).

VAN ROSSUM, Guido et al., 2007. Python Programming language. En: *Python Programming language. USENIX annual technical conference*. Vol. 41, págs. 1-36. N.º 1 (vid. pág. 12).

VENTURA BAUTISTA, Jesús Natividad, 2021. JQUERY, AJAX Definición de jQuery, programar JavaScript con jQuery. Selección de un elemento del documento mediante el id. Selección de elementos por el tipo de elementos. Método text (), text (valor). Métodos attr (nombre de propiedad), attr (nombre de propiedad, valor) y removeAttr (nombre de propiedad). Métodos addClass y removeClass. Método html () y html (valor). Administración de eventos con jQuery. Eventos mouseover y mouseout. Tipos de eventos, efectos, tipos efectos, iteración, Ajax, funciones anónimas, ejemplos y aplicaciones (vid. pág. 13).

VIDAL-SILVA, Cristian L; SÁNCHEZ-ORTIZ, Aurora; SERRANO, Jorge y RUBIO, José M, 2021. Experiencia académica en desarrollo rápido de sistemas de información web con Python y Django. *Formación universitaria*. Vol. 14, n.º 5, págs. 85-94 (vid. pág. 13).

Generado con LaTeX: 24 de noviembre de 2023: 12:43am

Apéndices

.1. Modelo de encuesta aplicada a especialistas

Tabla 9. Encuesta para aplicar Iadov

Estimado(a): Lea con cuidado cada pregunta antes de responder. Le agradecemos su participación y franqueza al decirnos honestamente lo que piensa sobre el uso de un Sistema para la gestión del proceso del Movimiento Deportivo en la UCI.
1- ¿Considera usted adecuado no usar las TIC para el proceso del Movimiento Deportivo ? _ No _ No sé _ Sí
2- ¿Satisfacen sus necesidades como Especialista de la Cultura Física, el sistema propuesto? _ No _ No sé _ Sí
3- ¿Utilizaría usted la aplicación propuesta a la hora de gestionar el proceso del Movimiento Deportivo en la UCI? _ Me satisface mucho. _ No me satisface tanto. _ Me da lo mismo. _ Me disgusta más de lo que me satisface. _ No me satisface nada. _ No sé qué decir.
4- ¿Modificaría usted algún elemento del sistema propuesto para la gestión del proceso del Movimiento Deportivo? Argumente.
5- ¿Considera usted que el sistema propuesto permite una correcta gestión del proceso del Movimiento Deportivo? Argumente.

.1.1. Resultados de la encuesta aplicada a especialistas

- Pregunta 1. De los 10 encuestados el 100 % concidera que No.
- Pregunta 2. De los 10 encuestados, 9 de ellos concideran que el sistema propuesto si satisfacen sus necesidades.
- Pregunta 3. De los 10 encuestados, 9 de ellos concideran que le sasistace mucho el uso del sistema propuesto y el resto concidera que le da lo mismo.

Para las restantes preguntas de la encuesta el total de la muestra coincidió que las funcionalidades que ofrece el sistema permite una correcta gestión del proceso del Movimiento Deportivo, los argumentos principalmente se basaron en:

- El sistema permite divulgar con éxito las delegaciones deportivas premiadas en el deporte determinado.
- El sistema permite hacer el proceso auditable y comprobable
- El sistema permite incorporar al proceso tanto a la comunidad universitaria como a la sociedad
- El sistema ofrece un medio para la retroalimentación del proceso.

Sin embargo en referencia a la pregunta cuatro se observó, que a pesar de que las funcionalidades con las que cuenta el sistema, no son suficientes para la gestión del proceso, para algunos encuestados sería beneficioso incluirle otras como reportes estadísticos, aplicación de mensajería y sistema recomendador para actividades a los usuarios.

.2. Guía de observación para el análisis de los sistemas similares

Tabla 10. Guía de observación para estudio de soluciones (Fuente: Elaboración propia).

No.	Indicadores	D	R	B	MB	E
1	Gestionan Eventos Deportivos.					
2	Ven a las actividades extensionistas como núcleo del proceso de extensión universitaria.					
3	Buscan promover el deporte en su más amplio significado.					
4	Permiten intercambio entre los interesados para la mejora del proceso.					
5	Proveen un medio para medir el impacto de la realización del proceso.					
6	Da visibilidad al proceso de forma tal que facilite la participación de los interesados en el mismo					
7	Permiten obtener las opiniones de los interesados sobre la realización del proceso.					

Leyenda	Deficiente	Regular	Bien	Muy bien	Excelente
	D	R	B	MB	E

- Web social para actividades deportivas (1)
- Sistema para la gestión de actividades estudiantiles en la residencia (2)
- Sistema de Gestión de la Nueva Universidad (SIGENU) (3)
- Sistema para el área de Extensión Cultural de la Universidad de las Ciencias Informáticas (4)
- Sistema para la gestión del Movimiento Deportivo (5)

.3. Historias de usuario

Tabla 11. Historia de usuario # 3

Historia de usuario	
Número: 3	**Nombre:** Listar Delegación Deportiva

Continúa en la próxima página

Tabla 11. Continuación de la página anterior

Usuario: Responsable de actividad	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capaz de listar las delegaciones deportivas que se crean.	
Observaciones: Este proceso es visible para cualquier usuario sin necesidad de estar autenticado.	
Interfaz:	

Nombre de la delegación	Premio Colectivo	Deporte	Entrenador
Peloteros	Oro	Beisbol	airelysm
Los Matadores	Bronce	Voleibol	airelysm
Los Tirimacos	No	Voleibol	airelysm

Página de .

Tabla 12. Historia de usuario # 4

Historia de usuario	
Número: 4	**Nombre:** Modificar Delegación Deportiva
Usuario: Responsable,solicitante,jefe de extensión IES, jefe extensión área	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** No. 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir modificar una Delegación Deportiva a través de un formulario donde el usuario sustituye los campos que necesite guardar con una nueva información.Para ello debe estar autenticado y contar con el rol requerido. • Nombre: Campo de texto (admite caracteres especiales y números). • Deporte: Campo de Selección. • Género: Campo de selección. • Facultad: Campo de selección. • Premio Colectivo: Campo de selección. • Entrenador: Campo de selección. • Deportistas: Campo de selección.	
Observaciones: La UI solo aparecerá si el usuario está autenticado. No se pueden dejar campos vacíos y los campos nombre y entrenador deben ser escritos con letra inicial mayúscula, en caso contrario el sistema debe mostrar un mensaje de error que avise al usuario el tipo de error y el lugar dónde se cometió.	

Continúa en la próxima página

Tabla 12. Continuación de la página anterior

Interfaz:

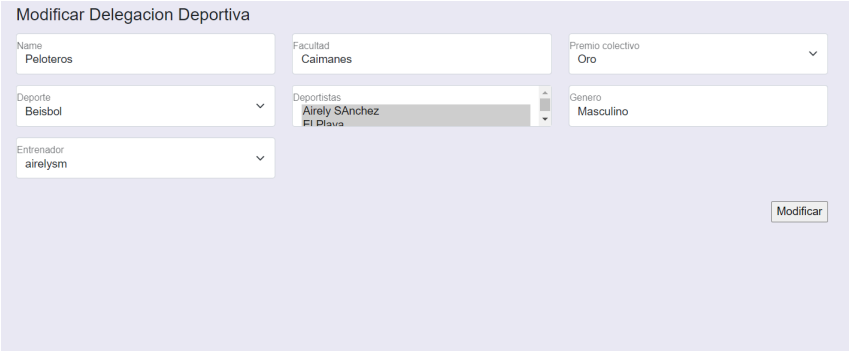


Tabla 13. Historia de usuario # 5

Historia de usuario	
Número: 5	**Nombre:** Mostrar Delegación Deportiva
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Bajo
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz de mostrar una delegacion deportiva determinada a partir de los detalles de la misma.	
Observaciones: El proceso es visible para cualquier usuario sin necesidad de estar autenticado.	
Interfaz:	

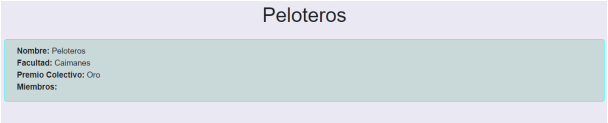


Tabla 14. Historia de usuario # 6

Historia de usuario	
Número: 6	**Nombre:** Eliminar Delegación Deportiva
Usuario: Responsable,solicitante,jefe de extensión IES, jefe extensión área	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Alto
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	

Continúa en la próxima página

Tabla 14. Continuación de la página anterior

Descripción: El sistema debe permitir eliminar una delegación deportiva determinada dentro del listado de las mismas.
Observaciones: El proceso solo se mostrará para el usuario con los roles requeridos por el sistema
Interfaz:

Tabla 15. Historia de usuario # 7

Historia de usuario	
Número: 7	**Nombre:** Buscar Delegación Deportiva
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir la búsqueda de delegaciones deportivas a partir del nombre de estas. El usuario hace click sobre la barra de búsqueda e introduce el nombre deseado.	
Observaciones: Este proceso sera visible para cualquier usuario autenticado dentro del sistema	
Interfaz:	

Tabla 16. Historia de usuario # 8

Historia de usuario	
Número: 8	**Nombre:** Asignar Premio a la Delegación Deportiva
Usuario: Responsable, solicitante	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Bajo
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir asignar un premio a una delegación deportiva determinada a partir de sus resultados en caso de ser una delegació de deporte colectivo.	
Observaciones: El proceso solo podrá ser ejecutado por los usuarios autorizados a partir de sus roles .	

Continúa en la próxima página

Tabla 16. Continuación de la página anterior

Interfaz:

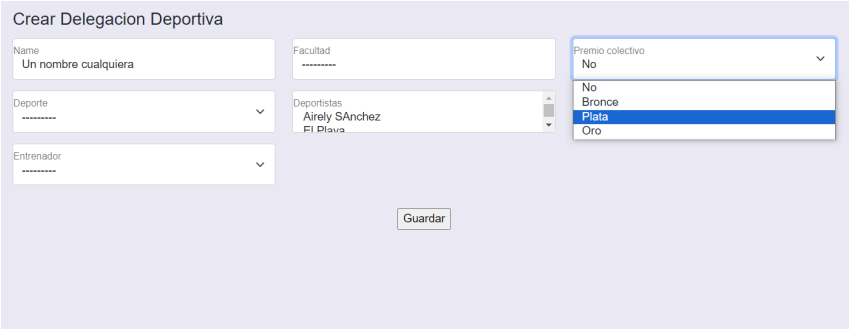


Tabla 17. Historia de usuario # 9

Historia de usuario	
Número: 9	**Nombre:** Modificar Premio a la Delegación Depotiva
Usuario: Responsable, solicitante	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Bajo
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir modificar un premio a una delegación deportiva determinada a partir de sus resultados en caso de ser una delegació de deporte colectivo.	
Observaciones: Este proceso solo lo podrá efectuar los usuarios autenticados y con el rol requerido.	
Interfaz:	

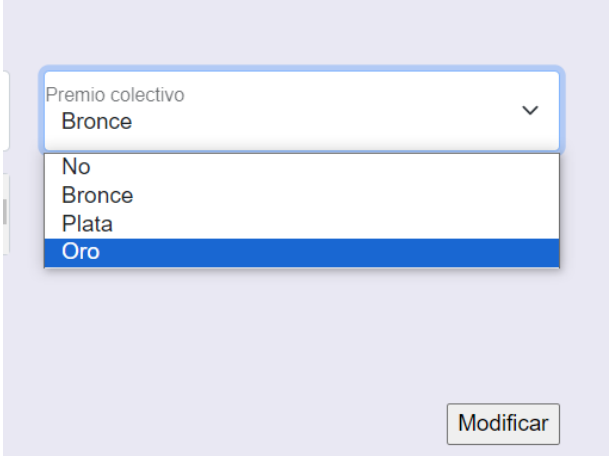


Tabla 18. Historia de usuario # 10

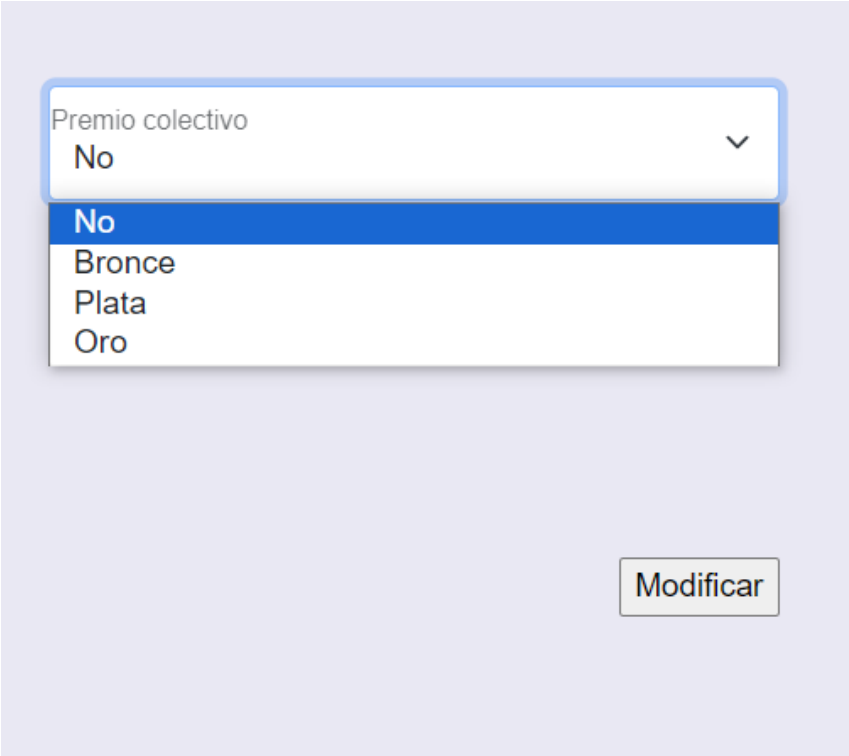
Historia de usuario	
Número: 10	**Nombre:** Eliminar premio de la delegación deportiva
Usuario: Responsable, solicitante	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Bajo
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir eliminar el premio colectivo de una delegación deportiva determinada.	
Observaciones: Este proceso lo puede hacer solo el usuario autenticado y con el rol requerido.	
Interfaz: 	

Tabla 19. Historia de usuario # 11

Historia de usuario	
Número: 11	**Nombre:** Mostrar Evento Deportivo
Usuario: Todos los usuarios	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	

Continúa en la próxima página

Tabla 19. Continuación de la página anterior

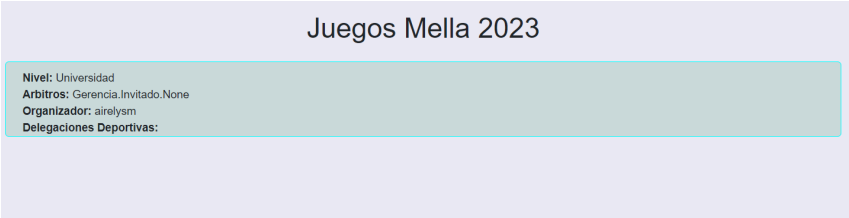
Descripción: El sistema debe ser capáz de mostrar los detalles de un evento deportivo seleccionado
Observaciones: Este proceso lo puede hacer cualquier usuario.
Interfaz: 

Tabla 20. Historia de usuario # 12

Historia de usuario	
Número: 12	**Nombre:** Crear Evento Deportivo
Usuario: Solicitante, Responsable, Director de Extensión	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Alto
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir crear un evento deportivo a partir de un formulario. • Nombre: Campo de texto (admite caracteres especiales y números). • Nivel: Campo de Selección. • CantDeportes: Campo numérico. • Organizador: Campo de selección. • FechaIni: Campo de selección. • FechaFin: Campo de selección. • Delegaciones Deportivas: Campo de selección. • Arbitros: Campo de selección.	
Observaciones: Esta operación la puede realizar solamente el usuario responsable del evento, el solicitante antes de ser confirmada la actividad por el responsable y el director de Extensión.	

Continúa en la próxima página

Tabla 20. Continuación de la página anterior

Interfaz:

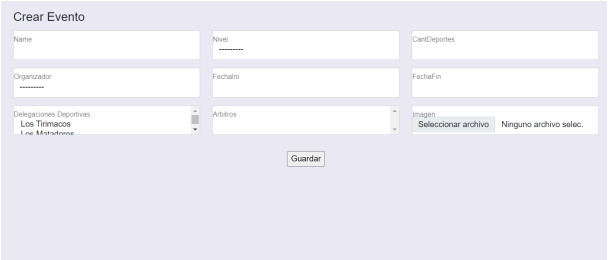


Tabla 21. Historia de usuario # 13

Historia de usuario	
Número: 13	**Nombre:** Listar Evento Deportivo
Usuario: Usuario común	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir listar todos los eventos deportivos creados.	
Observaciones: Esta operación la puede realizar cualquier usuario sin necesidad de estar autenticado.	
Interfaz:	

Tabla 22. Historia de usuario # 14

Historia de usuario	
Número: 14	**Nombre:** Modificar Evento Deportivo

Continúa en la próxima página

Tabla 22. Continuación de la página anterior

Usuario: Responsable, Director de Extensión	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Alto
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz de modificar un evento deportivo determinado a partir de un formulario. • Nombre: Campo de texto (admite caracteres especiales y números). • Nivel: Campo de Selección. • CantDeportes: Campo numérico. • Organizador: Campo de selección. • FechaIni: Campo de selección. • FechaFin: Campo de selección. • Delegaciones Deportivas: Campo de selección. • Arbitros: Campo de selección.	
Observaciones: Esta operación la puede realizar solamente los usuarios autenticados con el rol requerido.	
Interfaz:	

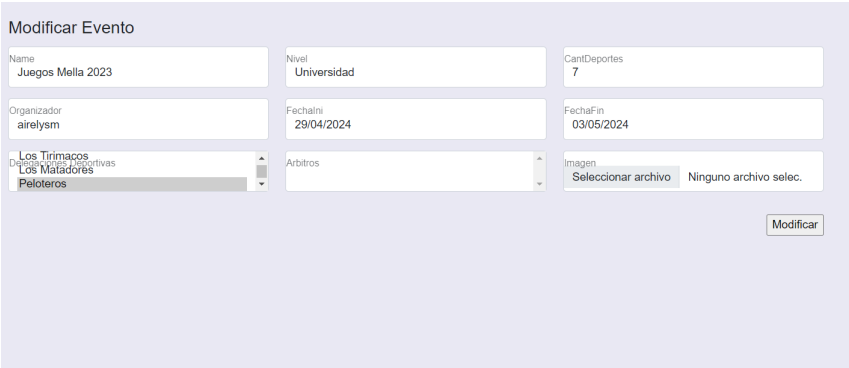


Tabla 23. Historia de usuario # 15

Historia de usuario	
Número: 15	**Nombre:** Eliminar Evento Deportivo
Usuario: Responsable, Director de Extensión	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir la eliminación de un evento deportivo determinado.	
Observaciones: Este proceso lo puede hacer solamente los usuarios autenticados con el rol requerido.	

Continúa en la próxima página

Tabla 23. Continuación de la página anterior

Interfaz:

Tabla 24. Historia de usuario # 16

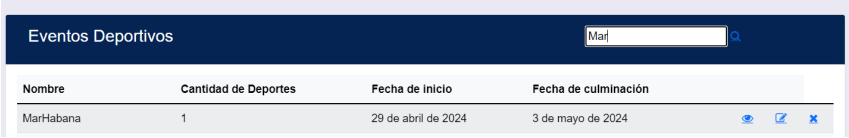
Historia de usuario	
Número: 16	**Nombre:** Buscar Evento Deportivo
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir a un usaurio realizar la búsqueda de un evento deportivo a seleccionar el campo de búsqueda e introducir el nombre deseado.	
Observaciones: El proceso puede ser ejecutado por cualquier usuario en el sistema.	
Interfaz: 	

Tabla 25. Historia de usuario # 17

Historia de usuario	
Número: 17	**Nombre:** Incertar Evidencia (multimedia) del Evento Deportivo
Usuario: Responsable, Director de Extensión	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Alto
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir la agregación de evidencias a partir de la ejecución de un campo de selección el cual accede al explorador de archivos del ordenador.	
Observaciones: Este proceso solo podra ser ejecutado por los usuarios autenticados con el rol requerido.	

Continúa en la próxima página

Tabla 25. Continuación de la página anterior

Interfaz:

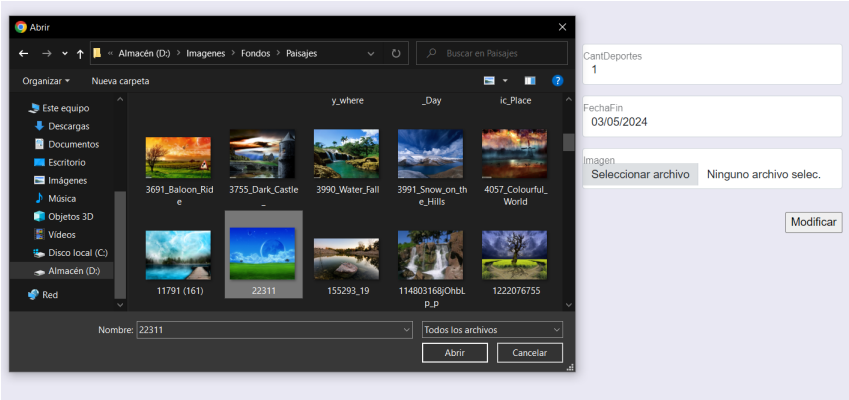


Tabla 26. Historia de usuario # 18

Historia de usuario	
Número: 18	**Nombre:** Eliminar evidencia (Multimedia)
Usuario: Director de Extensión, Responsable	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz de efectuar la eliminación de la evidencia guardada de un determinado evento deportivo.	
Observaciones: Este proceso solo podra ser ejecutado por los usuarios autenticados con el rol requerido.	

Continúa en la próxima página

Tabla 26. Continuación de la página anterior

Interfaz:

Tabla 27. Historia de usuario # 19

Historia de usuario	
Número: 19	**Nombre:** Mostrar Evidencia (Multimedia) Del Evento Deportivo
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz de mostrar la evidencia de un evento deportivo determinado.	
Observaciones: Este proceso puede ser visible para cualquier usuario.	

Continúa en la próxima página

Tabla 27. Continuación de la página anterior

Interfaz:

Tabla 28. Historia de usuario # 20

Historia de usuario	
Número: 20	**Nombre:** Listar Deportistas
Usuario: Usuario Común	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capaz de listar todos los deportistas que vayan siendo registrados.	
Observaciones: Este proceso podra ser visible para cualquier usuario en el sistema	

Interfaz:

Tabla 29. Historia de usuario # 21

Historia de usuario	
Número: 21	**Nombre:** Modificar Deportista
Usuario: Responsable, Director de Extensión	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Alto
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz modificar a un deportista determinado a través de un formulario. • Nombre y Apellidos: Campo de texto.Admite caracteres especiales, letras y números. • Edad: Campo de selección numérico, admite del 1 al 100. • Estado: Campo de selección. • Facultad: Campo de selección. • Deporte: Muestra el listado existente y el Campo de selección para añadir deportes. • Género: Campo de selección. • Medalla: Campo de selección.	
Observaciones: El proceso solo podrá ser ejecutado por los usuarios autenticados con los roles requeridos.	
Interfaz:	

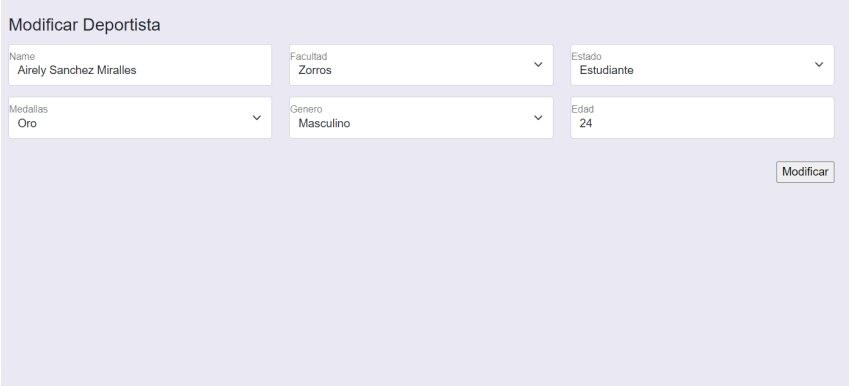


Tabla 30. Historia de usuario # 22

Historia de usuario	
Número: 22	**Nombre:** Eliminar Deportista
Usuario: Responsable, Director de Extensión	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz de realizar la eliminación de un deportista determinado.	

Continúa en la próxima página

Tabla 30. Continuación de la página anterior

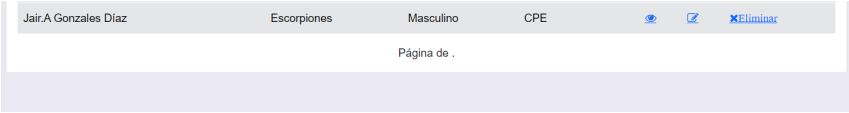
Observaciones: Este proceso solo podrá ser ejecutado por los usuarios autenticados con los roles requeridos.
Interfaz: 

Tabla 31. Historia de usuario # 23

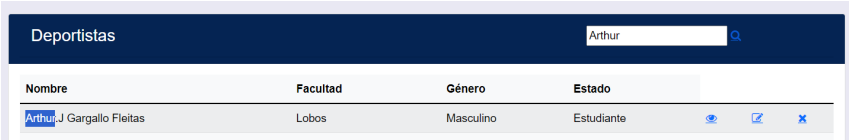
Historia de usuario	
Número: 23	**Nombre:** Buscar Deportista
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 16
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe permitir al usuario realizar la búsqueda de un deportista mediante la introducción de un nombre determinado dentro del campo de texto para buscar.	
Observaciones: El proceso lo podrá realizar cualquier usuario	
Interfaz: 	

Tabla 32. Historia de usuario # 24

Historia de usuario	
Número: 24	**Nombre:** Mostrar Detalles del Deportista
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Medio
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capáz de mostrar los detalles de un deportista determinado al ser seleccionado.	
Observaciones: El proceso será visible para cualquier usuario.	

Continúa en la próxima página

Tabla 32. Continuación de la página anterior

Interfaz:

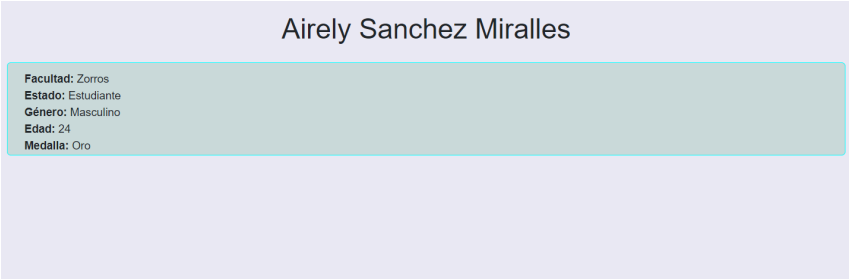


Tabla 33. Historia de usuario # 25

Historia de usuario	
Número: 25	**Nombre:** Mostrar Cronograma Deportivo
Usuario: Usuario común	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ser capaz de mostrar el cronograma deportivo según los eventos registrados ordenados por fecha de manera ascendente.	
Observaciones: Este proceso será visible para cualquier usuario	
Interfaz:	

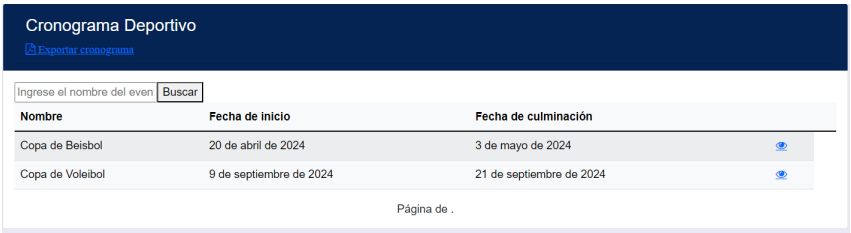


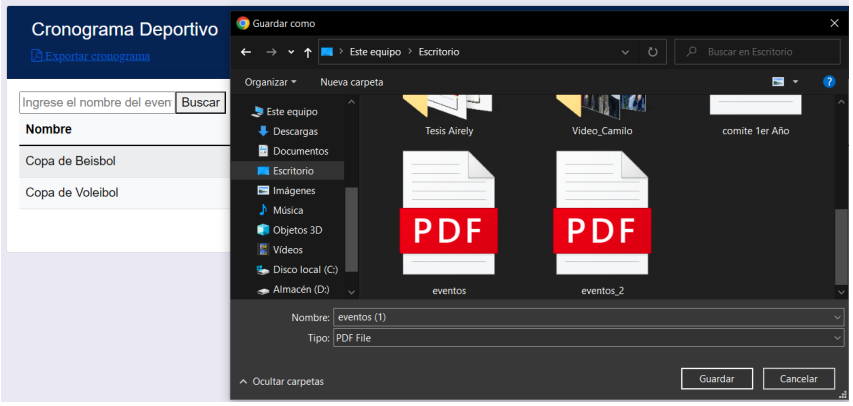
Tabla 34. Historia de usuario # 26

Historia de usuario	
Número: 26	**Nombre:** Buscar Evento Específico en el Cronograma
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	

Continúa en la próxima página

Tabla 34. Continuación de la página anterior

Descripción: El sistema debe permitir buscar dentro del cronograma de eventos registrados, un determinado evento.
Observaciones: Esta acción la podrá efectuar cualquier usuario en el sistema .
Interfaz:

Tabla 35. Historia de usuario # 27

Historia de usuario	
Número: 27	**Nombre:** Generar PDF del Cronograma Deportivo
Usuario: Usuario común	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Media
Puntos estimados: 1	**Iteración asignada:** 1
Programador responsable: Airely Sanchez Miralles	
Descripción: El sistema debe ofrecer la funcionalidad de poder exportar y guardar en formato de PDF un documento en el que quede registrado el cronograma deportivo arrojado por el módulo	
Observaciones: Esta función es accesible para cualquier usuario en el sistema.	
Interfaz: