



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 1

SISTEMA INFORMÁTICO PARA LA GESTIÓN DEL MOVIMIENTO DE ARTISTAS AFICIONADOS EN LA UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS.

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

Autor: Alejandro Campmany Cejas

Tutores: MSc. Julio César Espronceda Pérez

Ing. Álvaro Alejandro Acosta Ruiz

La Habana, 2022

Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber.

Albert Einstein

Dedicatoria

Dedico de forma especial esta tesis a mis padres por guiarme siempre en la dirección correcta, por brindarme su apoyo incondicional en todos los obstáculos a lo largo de los años ayudándome a ser cada día mejor persona.

Agradecimientos

A mis padres por motivarme a optar por una carrera universitaria, jamás podré pagarle lo mucho que me han dado, sin ellos no hubiese llegado hasta aquí.

A mis tías por haberme apoyado siempre y ser importantes pilares en mi vida.

A mi hermana por ayudarme en cada proyecto que me he planteado y siempre hacer un tiempo para mis problemas.

A mis abuelos por su cariño y constante preocupación.

A mis tutores, por apoyarme en todo el proceso del trabajo de diploma por instruirme y guiarme para lograr esta tesis, estoy eternamente agradecido.

A cada profesor que hizo su aporte en este proceso y aportarme tanto a lo largo de mi carrera.

A mi mejor amigo Lázaro por haberme apoyado en los estudios durante diez años y su ayuda incondicional.

A mis compañeros de clases por soportarme durante toda la carrera, por los buenos momentos, gracias.

A todos los que han contribuido a que logre este título universitario.

A todos muchas gracias.

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 ____ días del mes de _____ del año _____.

Alejandro Campmany Cejas
Autor

MSc. Julio César Espronceda Pérez
Tutor

Ing. Álvaro Alejandro Acosta Ruiz
Tutor

Las universidades son instituciones encargadas de formar profesionales integrales con un alto nivel cultural. La Extensión Universitaria (EU) es el proceso encargado de promover la cultura en la comunidad intra y extrauniversitaria. La gestión del Movimiento de Artistas Aficionados (MAA) es uno de los subprocesos de la extensión, el cual se encarga de realizar actividades y festivales con la finalidad de promover las distintas manifestaciones del arte en las universidades. En la Universidad de las Ciencias Informáticas (UCI) la gestión del MAA esta dirigido por la Dirección de Extensión, con el apoyo de la Federación Estudiantil Universitaria (FEU). Dicha gestión posee diferentes problemas que inciden en la realización de las actividades comprendidas dentro de la gestión del MAA. La presente investigación tiene como principal objetivo desarrollar un sistema informático que contribuya a la gestión de información del MAA. Para el desarrollo de la propuesta se realizó un estudio de sistemas homólogos. Se selecciono AUP en su variante UCI como metodología de desarrollo. Se utilizó Python como lenguaje de programación y Django como framework. Se realizó un análisis del negocio y se modelaron todos los procesos que realiza para obtener los requisitos. La aplicación permite gestionar el MAA, garantizando el control de la información y el acceso a la misma. Para garantizar la calidad de la solución se realizó una estrategia de pruebas. Para comprobar el nivel de satisfacción de los usuarios se realizó la técnica de Iadov, arrojando resultados positivos.

Palabras clave: cultura, extensión, gestión, sistema informático, universitaria festival.

Introducción	1
1 Marco teórico referencial del sistema para la gestión del Movimiento de Artistas Aficionados de la UCI	6
1.1 Conceptos asociados a la investigación	6
1.1.1 Extensión universitaria como proceso	6
1.1.2 Dirección de Extensión Universitaria	7
1.1.3 Movimiento de Artistas Aficionados	8
1.1.4 Festival de Artistas Aficionados	9
1.1.5 Gestión de información	9
1.1.6 Sistema de gestión de información	9
1.2 Estudio de sistemas homólogos	10
1.2.1 Ámbito internacional	10
1.2.2 Ámbito nacional	11
1.2.3 Consideraciones del estudio de sistemas homólogos	12
1.3 Metodologías, herramientas, y tecnologías	12
1.3.1 Método de Ingeniería de Software	12
1.3.2 Modelado de software	13
1.3.3 Lenguajes de programación	13
1.3.4 Framework de desarrollo	14
1.3.5 Bibliotecas	14
1.3.6 Sistema gestor de base de datos	15
1.3.7 Herramientas y tecnologías	15
1.3.8 Herramientas para control de versiones	15
1.3.9 Conclusiones del capítulo	16
2 Características del sistema para la gestión del Movimiento de Artistas Aficionados de la UCI	17
2.1 Descripción de la propuesta de solución	17
2.2 Especificación de requisitos del sistema	18
2.2.1 Requisitos funcionales	19

2.2.2	Requisitos no funcionales	19
2.3	Descripción de las historias de usuario	20
2.4	Estilo arquitectónico	23
2.4.1	Patrón arquitectónico	23
2.5	Diagrama de clases de diseño	24
2.6	Patrones de diseño	25
2.7	Modelo de datos	27
2.7.1	Descripción de las tablas	27
2.8	Conclusiones del capítulo	28
3	Implementación y prueba del sistema para la gestión del Movimiento de Artistas Aficionados de la UCI	29
3.1	Estándares de codificación	29
3.2	Diagrama de despliegue	30
3.3	Pruebas de software	31
3.3.1	Estrategia de pruebas	31
3.3.2	Pruebas unitarias	32
3.3.3	Pruebas funcionales	34
3.3.4	Pruebas de seguridad	38
3.3.5	Pruebas de rendimiento	40
3.3.6	Pruebas aceptación	41
3.4	Satisfacción de los usuarios	41
3.5	Consideraciones parciales	43
	Conclusiones	45
	Recomendaciones	46
	Referencias bibliográficas	47
	Apéndices	50
.1	Guía de observación para el análisis de los sistemas similares	51
.2	Modelo de encuesta aplicada	51
.2.1	Resultados de las entrevistas	52
.3	Historias de usuario	52

Índice de figuras

1.1	Proceso de Gestión de la Extensión Universitaria	8
2.1	Estilo arquitectónico Model-Template-View	24
2.2	Diagrama de clases	24
2.3	Modelo de Datos	27
3.1	Diagrama de Despliegue	30
3.2	Resultados de Pruebas (Elaboración Propia)	38
3.3	Resultados de las pruebas de seguridad, primera iteración	39
3.4	Resultados de las pruebas de seguridad, segunda iteración	40
3.5	Resultados de la prueba de carga	40
3.6	Resultados de la prueba de carga	41

Índice de tablas

1	Operacionalización de la variable independiente (Fuente: propia).	3
2	Operacionalización de la variable dependiente (Fuente: propia).	4
2.1	Requisitos funcionales. (Fuente: Elaboración propia).	19
2.2	Historia de usuario # 1	21
2.3	Historia de usuario # 2	22
3.1	Estrategia de pruebas. (Fuente: Elaboración propia).	31
3.2	Cálculo de la complejidad ciclomática del método Crear unidad artística	32
3.3	Caminos del grafo de flujo (Fuente: Elaboración propia).	33
3.4	Caso de Prueba para el camino básico 1 (Fuente: Elaboración propia).	34
3.5	Caso de prueba “Crear festival” (Fuente: Elaboración Propia).	35
3.6	Variables de caso de prueba “Crear festival”(Fuente: Elaboración Propia).	37
3.7	Cuadro Lógico de Iadov (Fuente: Elaboración propia).	42
8	Guía de observación para estudio de soluciones (Fuente: Elaboración propia).	51
9	Encuesta.	51
10	Historia de usuario # 3	52
11	Historia de usuario # 4	53
12	Historia de usuario # 5	54
13	Historia de usuario # 6	55
14	Historia de usuario # 7	56
15	Historia de usuario # 8	56
16	Historia de usuario # 9	57
17	Historia de usuario # 10	57
18	Historia de usuario # 11	58
19	Historia de usuario # 12	59
20	Historia de usuario # 13	59
21	Historia de usuario # 14	60
22	Historia de usuario # 15	60
23	Historia de usuario # 16	61
24	Historia de usuario # 17	61

25	Historia de usuario # 18	62
26	Historia de usuario # 19	63
27	Historia de usuario # 20	63
28	Historia de usuario # 21	64
29	Historia de usuario # 22	65
30	Historia de usuario # 23	65
31	Historia de usuario # 24	66
32	Historia de usuario # 25	66
33	Historia de usuario # 26	67
34	Historia de usuario # 27	67

La introducción y el uso de las Tecnologías de la Información y las Comunicaciones y de Internet (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. La Informatización de la Sociedad 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 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. 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 (González; Ramón y Mercedes, 2001)

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 (Martínez Cabrera, 2019). Las cuales son organizadas desde la Vicerrectoría de Extensión Universitaria en coordinación con las facultades docentes y las organizaciones estudiantiles, contando con la colaboración de profesores e instructores de danza, música, teatro, artes plásticas y literatura quienes imparten talleres de creación y apreciación.

El Festival de Artistas Aficionados (FAA) de la FEU es el momento cumbre del MAA. Movimiento que se comenzó a organizar en Cuba a partir de la década del sesenta, con una amplia participación de los diferentes sectores estudiantiles: Pioneros, Federación de Estudiantes de la Enseñanza Media (FEEM), FEU y laborales, a través de los sindicatos. En él se premian no solo la calidad de las obras en concurso, sino también el esfuerzo y la dedicación de aficionados, profesores, directivos e instructores de arte, sin los cuales no fuera posible exhibir el prestigio que le ha merecido el respaldo incondicional del Ministerio de Cultura y su red de consejos, instituciones y organizaciones culturales (MES, 2022).

En la UCI, como parte de la infraestructura para garantizar el desarrollo de las artes se encuentra el centro Wifredo Lam, dirigido por la Dirección de Extensión Cultural. Este dispone de aulas especializadas para teatro, danza y música. Además, cuenta con una sala de video, galería de artes plásticas, librería, complejo recreativo, plazas y teatros, brindando así la calidad requerida para el desarrollo de los festivales (UCI, s.f.) En la realización de los festivales participan artistas en diferentes manifestaciones del arte, de las presentaciones se registran sus datos. En sus presentaciones se recogen datos tales como los resultados obtenidos, el nombre de la presentación y evidencias que son almacenadas y pueden ser utilizadas en posteriores realizaciones de festivales. Además, son necesarias en caso de que dichos artistas cualifiquen para participar en eventos de artistas aficionados 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 presentaciones premiadas. Las presentaciones se graban en medios digitales para ser preservadas y en ocasiones ser difundidas en medios digitales que poseen las universidades. Esta información es llevada de forma manual; lo que ocasiona que se realice doble a causa de pérdidas. Requiere de la inversión de tiempo, recursos humanos y materiales. Se ve agudizada al guardar la misma en archivos digitales, 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 manifestación.

Analizando el proceso de gestión del MAA se encontraron las siguientes limitaciones:

- La información del MAA se encuentra desorganizada y no está almacenada de forma centralizada provocando que haya pérdida de información y en ocasiones duplicidad de la misma.
- No se aprovecha la infraestructura tecnológica de la universidad y las bondades de las TIC para la gestión del MAA afectando la divulgación de las actividades, retrasos en la entrega de información y falta de recursos en la concepción del trabajo.
- No se garantiza el acceso a la información generada del proceso de gestión del MAA, lo que provoca dificultades a la hora de trabajar estadísticas, evaluar desarrollo y dar seguimiento al trabajo con aficionados.
- Poca efectividad en la gestión de la información generada del MAA que provoca afectaciones en los procesos de gestión de artistas y agrupaciones.
- Pérdida de evidencias de las actividades realizadas vinculadas al MAA.
- No existe lugar donde se publiquen las premiaciones de los festivales provocando el desconocimiento

de dicha información por artistas y la comunidad universitaria.

La problemática planteada conlleva al siguiente **problema investigación**: ¿Cómo contribuir a la gestión del Movimiento de Artistas Aficionados en la Universidad de las Ciencias Informáticas? Se define como **objeto de estudio** los procesos de gestión de información. Teniendo como **objetivo general** desarrollar un sistema informático que contribuya a la gestión del Movimiento de Artistas Aficionados en la Universidad de las Ciencias Informáticas. Permitiendo identificar como **campo de acción** a los procesos de gestión de la información del Movimiento de Artistas Aficionados en la UCI.

Para llevar a cabo el cumplimiento del objetivo se plantean las siguientes **tareas de la investigación**:

- Estudio de los principales conceptos asociados al tema.
- Estudio y análisis de trabajos relacionados con los sistemas de gestión de la información.
- Diagnóstico del proceso de gestión del MAA.
- Estudio de la metodología y las tecnologías a emplear en el proceso de desarrollo de software.
- Análisis y diseño del sistema para la gestión de la información del MAA.
- Implementación del sistema para la gestión de la información MAA.
- Validación y prueba del sistema informático para la gestión de la información del MAA.

La investigación se realiza teniendo como **hipótesis** que: el desarrollo de un sistema informático para la Dirección de Extensión Universitaria debe contribuir a la gestión de información del Movimiento de Artistas Aficionados en la Universidad de las Ciencias Informáticas.

En las siguientes tablas se presenta la operacionalización de las variables independiente y dependiente respectivamente.

Tabla 1. Operacionalización de la variable independiente (Fuente: propia).

Variable independiente	Indicadores	Unidad de medición
Sistema informático	Funcionalidad	Alta: Todos los subprocesos Media: Al menos 4 subprocesos Baja: Menos de 4 subprocesos.
	Interfaz de usuario	5: Botones coherentes. 3: Errores ortográficos. 1: Errores ortográficos y botones incoherentes.
	Nivel de satisfacción	5: Satisfacción clara. 4: Más satisfecho que insatisfecho. 3: No definida. 2: Más insatisfecho que satisfecho. 1: Clara insatisfacción

Tabla 2. Operacionalización de la variable dependiente (Fuente: propia).

Variable dependiente	Dimensión	Indicadores	Unidad de medida
Gestión de información	Organización	Información accesible únicamente por el personal autorizado.	Definición de roles.
		Información estructurada.	Utilización de identidad marcaría de la UCI.
	Centralización	Información almacenada de manera centralizada.	Diseño de la base de datos.
	Disponibilidad	Tiempo de consulta de la información en segundos.	Bien: Menos de 2 segundos. Regular: Hasta 4 segundos. Mal: Más de 5 segundos.

Para obtener los conocimientos necesarios, con la finalidad de hacer posible el cumplimiento del objetivo trazado en el trabajo, se utilizaron algunos de los **métodos científicos** existentes, tanto teóricos como empíricos.

Los **métodos teóricos** permiten estudiar las características del objeto de investigación que no son observables directamente, facilitando la construcción de modelos e hipótesis de investigación. De estos se utilizaron los siguientes:

- **Histórico-Lógico:** Permite realizar un estudio del arte sobre sistemas de gestión de información similares que se han hecho para obtener características y elementos fundamentales.
- **Modelación:** La modelación es método con el que se crean abstracciones para representar la realidad. Este método se utiliza para representar a través de diagramas la estructura, relaciones y características de la solución.
- **Análisis-Síntesis:** Se emplea para ampliar los conocimientos respecto al tema, mediante la consulta de bibliografía correspondiente para ser usada en el desarrollo del sistema en cuestión.

Los **métodos empíricos:** permiten estudiar las características del objeto de investigación que no son observables directamente, proviene de la experiencia y es sometido a determinada elaboración racional; de ellos se emplearon los siguientes:

- **Análisis documental:** Se emplea para el estudio de documentos clasificados, elementos teóricos que sustentan la concepción sobre la gestión de este proceso, la revisión de la bibliografía y de fuentes básicas de información.
- **Entrevista:** Se realizan entrevistas a trabajadores calificados del departamento de gestión universitaria para facilitar la comprensión del negocio, obtener información sobre la problemática existente, obtener ideas y referencias.
- **Observación:** Se emplea para obtener información necesaria de las soluciones que ya existen para la gestión del MAA.

El presente trabajo de diploma posee la siguiente estructura: introducción, tres capítulos, conclusiones generales, recomendaciones, referencias bibliográficas utilizadas durante el transcurso de la investigación y una serie de anexos que complementan el trabajo escrito. A continuación, se muestra una breve descripción de los capítulos:

- **Capítulo 1. Marco teórico referencial sobre el sistema para la gestión del Movimiento de Artistas Aficionados de la Universidad de las Ciencias Informáticas.** En este capítulo se presentan los conceptos y elementos teóricos para garantizar un mejor entendimiento de la situación problemática planteada. Se realiza un estudio de las herramientas y tecnologías para el desarrollo web. Se realiza un análisis de los sistemas de gestión de información y se definen los lenguajes de programación y las metodologías para el desarrollo web.
- **Capítulo 2. Características del sistema para la gestión del Movimiento de Artistas Aficionados de la Universidad de las Ciencias Informáticas.** En este capítulo se explica el proceso de desarrollo de software que conlleva a la solución. Se exponen los artefactos a tener en cuenta obtenidos según la metodología seleccionada tales como, requisitos funcionales y no funcionales, arquitectura del software y el modelo de datos. También se definen los patrones de diseño y de arquitectura a utilizar durante el desarrollo de la solución.
- **Capítulo 3. Implementación y prueba del sistema para la gestión del Movimiento de Artistas Aficionados de la Universidad de las Ciencias Informáticas.** En este capítulo se abordan aspectos relacionados con la implementación del sistema a desarrollar. Se realizan pruebas de software para validar la solución propuesta y garantizar que se cumplan las funcionalidades de las especificaciones requeridas.

Marco teórico referencial del sistema para la gestión del Movimiento de Artistas Aficionados de la UCI

En el presente capítulo se analizan los conceptos que resultarán útiles para un mejor entendimiento de los términos asociados al problema planteado, 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 EU en la UCI.

En la realización de una investigación es necesario realizar un estudio teórico y llevar en práctica lo aprendido. Se realiza una profunda búsqueda bibliográfica 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. Se realiza un análisis de la metodología, herramientas y tecnologías a emplear en el desarrollo de la propuesta de solución.

1.1. Conceptos asociados a la investigación

Para ayudar a entender el desarrollo de la investigación es necesario definir una serie de conceptos, los cuales brindan soporte para la fundamentación teórica de la misma.

1.1.1. Extensión universitaria como proceso

La 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 extrauniversitaria, la cual se articula con la formación y la investigación y junto con estas contribuye al desarrollo sociocultural y de los valores (Batista, 2016). 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.

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 posee varios campos como son el movimiento deportivo, el MAA, las cátedras honoríficas, los cursos de capacitación y los proyectos comunitarios. Contribuyendo a elevar el nivel cultural de la comunidad y al desarrollo de los estudiantes profesores y trabajadores.

Un proceso es una secuencia de pasos dispuesta con algún tipo de lógica que se enfoca en lograr algún resultado específico según el sitio Definición de (Definición.de, 2022). Según el diccionario de Oxford Languages, un proceso es el procesamiento o conjunto de operaciones a que se somete una cosa para elaborarla o transformarla (Languages, 2022).

El autor define proceso de extensión universitaria como 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.

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 (González; Ramón y Mercedes, 2001). 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 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; 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.

En la siguiente ilustración se muestra la interrelación que existe entre los procesos clave que desarrolla la universidad (formación, investigación y extensión universitaria) y los diferentes subprocesos de la extensión.

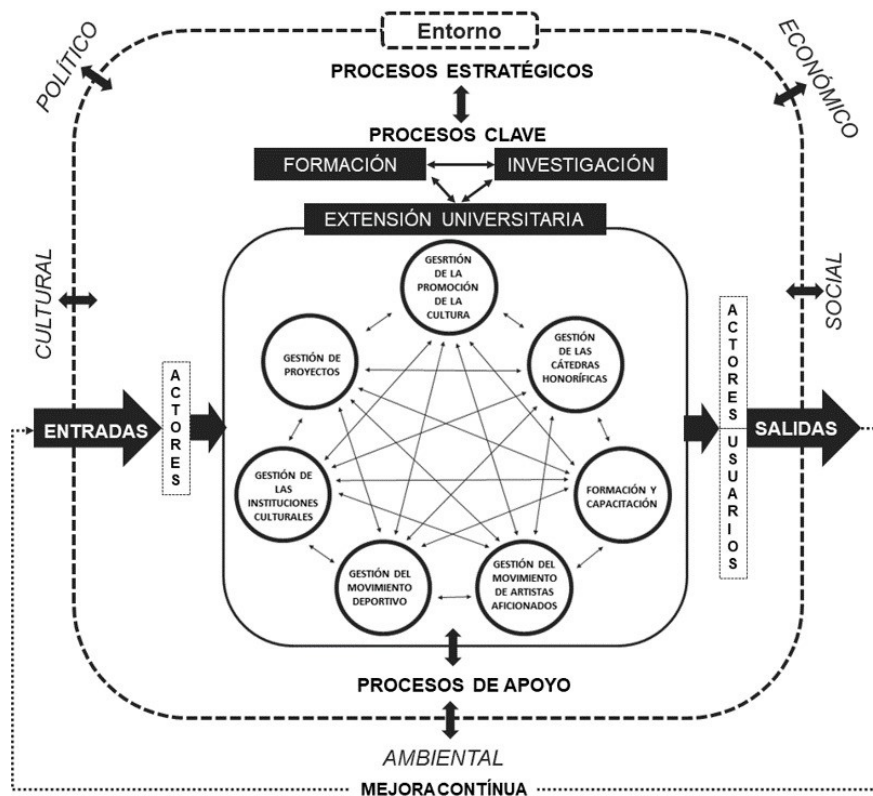


Figura 1.1. Proceso de Gestión de la Extensión Universitaria (fuente: (González Aportela; González Fernández-Larrea; Zambrano Llor; Balsinde Herrera y Batista Mainegra, 2022))

1.1.3. Movimiento de Artistas Aficionados

La UCI posee un MAA integrado por estudiantes y profesores que trabajan en conjunto en talleres de creación y apreciación tanto de danza, música, teatro, artes plásticas y literatura. Este esfuerzo se evidencia en las cotas de calidad que se proponen alcanzar con su arte en los FAA. Los festivales constituyen el momento cumbre del MAA, 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 MAA es reconocido por el Ministerio de Cultura, como uno de los de mayor nivel y calidad en el país, avalado por su sistematicidad y proyección social (Martínez Cabrera, 2019).

Para un artista aficionado poder participar en esta actividad en la UCI, debe presentar una planilla de inscripción con un conjunto de datos del mismo, como: nombre y apellidos, carnet de identidad, carrera, número de solapín, teléfono, correo electrónico y si es estudiante egresado.

1.1.4. Festival de Artistas Aficionados

Los festivales se realizan anualmente en todas las universidades del país. Comienzan desde la base que es el FAA a nivel de facultad, luego se realiza el FAA a nivel de universidad, en caso de haber más de una universidad en la provincia se realiza el FAA provincial y termina con el FAA a nivel nacional. Donde entre cada festival se realiza un proceso de selección con los ganadores del festival del nivel inferior inmediato con el fin de no llevar a escena muchas obras similares garantizar la calidad del festival en cuestión a realizar. El Festival Nacional de Artistas Aficionados de la FEU es considerado como la bienal cultural de la Educación Superior cubana, constituye el máximo peldaño de un ciclo competitivo que abarca dos cursos académicos. En él se premian no solo la calidad de las obras en concurso, sino también el esfuerzo y la dedicación de aficionados, profesores, directivos e instructores de arte, sin los cuales no fuera posible exhibir el prestigio que le ha merecido el respaldo incondicional del Ministerio de Cultura y su red de consejos, instituciones y organizaciones culturales (MES, 2022).

1.1.5. Gestión de información

Gestión de la información (GI) 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 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).

Se concluye que 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.

1.1.6. Sistema de gestión de información

Un sistema es un conjunto de elementos dinámicamente relacionados, en interacción que desarrollan una actividad para lograr un objetivo o propósito.

Un sistema de gestión de información son herramientas que se basadas en el diseño de gestión del conocimiento que propone la codificación del conocimiento explícito y la difusión y socialización del conocimiento tácito dentro de las organizaciones (M. D. Pérez, 2017). 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.

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 (SNIES, 2012).

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 MAA. Además, se gestiona la información de diferentes formas incluyendo en formato duro, siendo esto una de las problemáticas a resolver por la presente investigación.

- **Sistema para la Prevención de la Deserción en la Educación Superior:** El Ministerio de Educación Nacional de Colombia propuso disminuir la deserción estudiantil en la educación superior como parte de la estrategia planteada para aumentar la cobertura, la calidad y la eficiencia educativas. Entre otras acciones, puso en marcha el Sistema para la Prevención de la Deserción en la Educación Superior (Spadies) ante la necesidad de contar con una visión sectorial e integrada de la problemática de la deserción, a partir de la cual se pudiera disponer de una conceptualización, una medición y una metodología de seguimiento del fenómeno aplicables a todas las instituciones de educación superior de ese país (COLOMBIA, 2016).

Este sistema le permite al Ministerio medir y monitorear los factores determinantes de la deserción, conocer su evolución en el tiempo y ver cómo se comportan diferentes instituciones y regiones. Igualmente, hace posible que cada institución cuente hoy con un perfil de sus estudiantes y con sistemas de alertas tempranas sobre los factores que los hacen vulnerables, lo que sirve para orientar de forma más eficiente apoyos y políticas. Además, permite hacer seguimiento estadístico a los niveles de deserción, seguimiento a las condiciones de ingreso de los estudiantes, seguimiento al comportamiento de factores determinantes del fenómeno, estimación del riesgo de deserción para cada estudiante. Facilita la elección y evaluación de estrategias institucionales de apoyo a los estudiantes (ibíd.).

1.2.2. **Ámbito nacional**

- **Sistema automatizado para la gestión de la Extensión Universitaria:** Sitio desarrollado en la Universidad de Pinar del Río por un colectivo de profesores de dicha institución. El sitio es una aplicación informática que potencia el proceso de gestión de la EU de forma tal que favorece la promoción y el control de las actividades extensionistas. El sitio permite publicar información relacionada con los diferentes eventos que se van a estar efectuando, ya sean deportivos o culturales, mostrando en cada caso los deportes o categorías que los integran, la fecha en que se efectuarán, así como los participantes y los resultados que se obtienen. Posee un espacio dedicado a las noticias, lo cual permite a los usuarios mantenerse actualizados sobre los principales acontecimientos. Brinda la posibilidad, a cualquier usuario del sitio de lanzar diferentes convocatorias, las cuales van a ser publicadas con la finalidad de que estén disponibles para toda la comunidad estudiantil (Pacheco Surez Y., 2012).

Este sitio no responde a algunas de las necesidades de la universidad, entre las cuales se encuentra la gestión de la información de los festivales de artista aficionados. Además, presenta una sección para publicar noticias lo cual no está entre los problemas a resolver de la presente investigación. También gestiona la información en la residencia estudiantil, no siendo útil debido a la existencia de un sistema en la universidad que gestiona dicha información.

- **Sistema informático de gestión universitaria (SIGUA):** 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 (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, s.f.).

- **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).

Para la realización del sistema se realiza una búsqueda y análisis de la bibliografía existente. A través del análisis y la síntesis, se identifican los principales elementos y rasgos que caracterizan a los sistemas de gestión académica, los diferentes procesos de formación, así como el papel preponderante del personal de secretaría de la Universidad. Se utiliza la observación directa del sistema de trabajo de los profesores, secretarías docentes, la secretaría general y de todo el personal implicado en el proceso de formación y el flujo de información en las diferentes áreas de la universidad para detectar las debilidades y fortalezas en esta área (ibíd.).

Este sistema gestiona información acerca de las áreas docentes, productivas e investigativas de la universidad con respecto a la Residencia, Cooperación, Biblioteca, Investigación, Tecnologías, Teleformación, Ingreso y Egreso, así como Extensión, Pregrado y Postgrado. El cual fue desarrollado por la Dirección de Informatización perteneciente a la Vicerrectoría de nuestra universidad.

- **Sistema de Gestión Universitaria (SIGENU):** Es un sistema informático (si) 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 todos los Centros de Educación Superior (CES) adscriptos al MES, así como en el Instituto Superior de Diseño Industrial (ISDI) y en el Instituto Superior de Relaciones Internacionales (ISRI), el Fajardo y otros centros no adscriptos al MES. Todos los CES del país disponen de su matrícula de los cursos diurnos (CD) y cursos por encuentro (CPR) actualizada. Los servicios ya 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).

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. Se evidencia que en nuestros días no existe ningún sistema dedicado a la gestión de la información del MAA aplicable a la UCI. Por lo tanto, constituye una necesidad primordial para la universidad desarrollar un sistema que permita gestionar dicho proceso. También el estudio posibilitó conocer las tecnologías empleadas en el desarrollo de este tipo de sistema.

1.3. Metodologías, herramientas, y tecnologías

El presente epígrafe presenta las metodologías, herramientas y tecnologías que guían el proceso de desarrollo de la aplicación. Estas fueron impuestas por el cliente como requisitos de restricción de diseño (se aborda en el capítulo 2) debido a que se espera que el sistema se vincule a otro sistema mayor.

1.3.1. Método de Ingeniería de Software

Para el desarrollo de la solución se decide emplear la metodología Proceso Ágil Unificado (AUP, por sus siglas en inglés) en su variante UCI en el escenario 4 debido a que es la utilizada como estándar en el proceso productivo en dicha institución. El equipo de desarrollo es pequeño, el tiempo que se dispone para desarrollar la aplicación es corto y el cliente es cercano al equipo de desarrollo.

1.3.2. Modelado de software

- **Lenguaje Unificado de Modelado (UML, por sus siglas en inglés) 8.0:** El UML desempeña un rol importante no solo en el desarrollo de software, sino también en los sistemas que no tienen software en muchas industrias, ya que es una forma de mostrar visualmente el comportamiento y la estructura de un sistema o proceso. El UML ayuda a mostrar errores potenciales en las estructuras de aplicaciones, el comportamiento del sistema y otros procesos empresariales. Este lenguaje de modelado automatiza la producción de software y los procesos, simplifica las complejidades y ayuda a resolver los problemas arquitectónicos constantes (Team, 2019). El UML será empleado en la elaboración de la mayoría de los diagramas de la presente investigación.
- **Visual Paradigm 8.0:** Visual Paradigm es una herramienta de 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 completo 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 (De Arma-Hernández y Sablón-Fernández, 2019).

Entre sus principales características se encuentran:

- Disponibilidad en múltiples plataformas (Windows, Linux).
- Capacidades de ingeniería directa e inversa.
- Posee licencia gratuita y comercial.
- Soporta aplicaciones web.
- Permite realizar diagramas de flujo de datos.
- Generación de bases de datos. Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Editor de figuras (Pressman, 2002).

1.3.3. 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. Entre sus principales ventajas cabe mencionar:
 - 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.
 - Se puede utilizar fácilmente con otros lenguajes de programación conocidos.
 - 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.

- Se puede trasladar a través de diferentes sistemas operativos de computadora, como Windows, macOS, Linux y Unix.
 - Hay muchos recursos útiles disponibles en Internet si desea aprender Python.
 - Permite escribir un programa de Python con menos líneas de código en comparación con muchos otros lenguajes (Vidal-Silva; Sánchez-Ortiz; Serrano y Rubio, 2021).
- **JavaScript:** JavaScript es un robusto lenguaje de programación que se puede aplicar a un documento Lenguaje de Marcado de Hipertexto (HTML, por sus siglas en inglés) 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 (mozilla.org, 2022).
 - **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 (Mozilla, 2022). En la actualidad todas las páginas web están estructuradas con este lenguaje.
 - **Hojas de estilo de cascada (CSS, por sus siglas en inglés) 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.

1.3.4. 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 (Project, 2022).

1.3.5. 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 (getbootstrap, 2022). Es compatible con la mayoría de navegadores webs actuales. Es usado por muchas organizaciones destacadas.

- **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 XML (AJAX, por sus siglas en inglés) sean mucho más simples con una Application Programming Interfaces (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.

1.3.6. 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 Structured Query Language (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 (PostgreSQL, 2022).

1.3.7. Herramientas y tecnologías

- **Entorno de desarrollo integrado: Visual Studio Code:** Es un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés) 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 (Code, 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, HTML/CSS, JSON, PHP, entre otros; los cuales son muy usados en el desarrollo de aplicaciones web.

1.3.8. 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 (M. R. Rodríguez; Gamboa y S. T. López, 2019). Este sistema es de código abierto y gratuito. Entre sus principales características están:
 - Mantener un único repositorio de código fuente.
 - Automatizar la construcción del proyecto.
 - Autodiagnóstico de la construcción.
 - Entregar los cambios diariamente a la línea base.

- Construir la línea base tras cada entrega.
- Probar en una réplica del entorno de producción.
- Mantener una ejecución rápida de la construcción del proyecto.
- Fácil obtención del último ejecutable del proyecto.
- Publicar el estado del proyecto.
- Automatizar el despliegue.

1.3.9. 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:

- Se define como propuesta solución un sistema web que permita a los profesores de la DEU la gestión del MAA.
- 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 de lo acordado con el cliente se estableció como metodología de desarrollo AUP variante UCI en su escenario 4. Se escogió Django como framework de desarrollo, lenguajes de programación como Python, HTML, CSS y JavaScript. Así como el uso de herramientas tales como Visual Paradigm, Microsoft Visual Studio Code y PostgreSQL.

Características del sistema para la gestión del Movimiento de Artistas Aficionados de la UCI

El presente capítulo aborda las características del sistema de la propuesta de solución luego de haber realizado el estudio del arte. Sus requisitos funcionales y no funcionales. Se aborda el estilo arquitectónico y los patrones de diseño para las buenas prácticas. Se presenta el modelo de datos y los principales artefactos de ingeniería que propone Proceso Ágil Unificado en su variante UCI (AUP-UCI, por sus siglas en inglés) en su escenario 4 así como la descripción detallado de los mismos.

2.1. Descripción de la propuesta de solución

Teniendo en cuenta la situación problemática planteada se propone como solución un sistema de gestión de información del MAA. Dicho sistema permitirá tener la información centralizada y accesible. El sistema debe permitir tener un registro histórico de los FAA disponible para los usuarios.

Festivales

En esta sección se muestran los festivales registrados en el sistema accesible para todos los usuarios. Si el usuario esta autenticado permite filtrar por los festivales en los que ha participado dicho usuario. Si el usuario es del rol encargado y cuenta con los permisos adecuados tendrá acceso a gestionar la información de los festivales.

Unidades artísticas

Permite visualizar las unidades artísticas con sus integrantes registrados en el sistema. Si el usuario esta autenticado permite filtrar por las unidades artísticas de las que forma parte dicho usuario. Si el usuario es del rol encargado y cuenta con los permisos adecuados tendrá acceso a gestionar la información de las unidades artísticas. El sistema debe permitir generar la planilla de inscripción de la unidad artística y permitir su exportación.

Premios

En esta sección aparecen las presentaciones por festivales. Si el usuario pertenece al rol encargado podrá asignarle un premio. Cuando sea asignado un premio a una unidad artística debe generar una notificación en el sistema y una notificación por correo a los miembros de dicha unidad artística. Si es un usuario común solo podrá visualizar los premios. Al hacer click en una presentación debe redirigir a la actividad donde aparecen sus detalles y sus evidencias (imágenes o videos).

Programa artístico por manifestación

Esta sección del sistema permite filtrar las actividades por tipo (ensayo o presentación) y por manifestación para generar un cronograma. Debe permitir exportar el cronograma en formato pdf.

Roles

Los roles son la forma que tiene el sistema de garantizar que cada usuario realice solo las funcionalidades que tiene permitidas. El sistema contará con siete roles que se detallan a continuación:

- Usuario Anónimo: El usuario solo tendrá acceso a las funcionalidades de listar y mostrar información.
- Usuario Autenticado: Rol que solo podrá visualizar información, filtrar dicha información donde aparezca presente y valorar actividades.
- Encargado de facultad: Es el rol encargado de gestionar la información de las unidades artísticas, premios, cronogramas artísticos y festivales a nivel de facultad.
- Encargado de universidad: Es el rol encargado de gestionar la información de las unidades artísticas, premios, cronogramas artísticos y festivales a nivel de universidad.
- Encargado provincial: Es el rol encargado de gestionar la información de las unidades artísticas, premios, cronogramas artísticos y festivales en el nivel provincial.
- Encargado nacional: Es el rol encargado de gestionar la información de las unidades artísticas, premios, cronogramas artísticos y festivales en el nivel nacional.
- Admin: Rol encargado de gestionar el sistema y gestionar usuarios. Cuenta con la mayor cantidad de permisos.

2.2. Especificación de requisitos del sistema

Con el objetivo de desarrollar el sistema propuesto con las necesidades dictadas por el cliente final a partir del uso de la tormenta de ideas y la realización de talleres, se obtuvieron los requisitos funcionales y no funcionales.

2.2.1. Requisitos funcionales

Teniendo en cuenta las características de la propuesta en cuanto a prestaciones que debe tener el sistema se identifican 27 requisitos funcionales. A continuación, la tabla 2.1 muestra los requisitos funcionales del sistema.

Tabla 2.1. Requisitos funcionales. (Fuente: Elaboración propia).

No.	Nombre	Prioridad	Complejidad
RF1	Crear unidad artística	Alta	Media
RF2	Listar unidades artísticas	Baja	Baja
RF3	Modificar unidad artística	Media	Media
RF4	Eliminar unidad artística	Baja	Baja
RF5	Generar PDF de la planilla de la unidad artística	Media	Media
RF6	Mostrar detalles de la unidad artística	Baja	Baja
RF7	Buscar unidades artísticas	Baja	Baja
RF8	Mostrar dossier de actividades de la unidad artística	Baja	Baja
RF9	Generar PDF del dossier de actividades	Media	Media
RF10	Asignar premio a unidad artística	Alta	Media
RF11	Modificar premio a unidad artística	Media	Media
RF12	Eliminar premio a unidad artística	Baja	Baja
RF13	Listar premios	Baja	Baja
RF14	Buscar premios	Baja	Baja
RF15	Notificar premio por correo a los miembros de la unidad artística	Alta	Alta
RF16	Crear festival	Alta	Media
RF17	Listar festival	Baja	Baja
RF18	Modificar festival	Media	Media
RF19	Eliminar festival	Baja	Baja
RF20	Mostrar detalles del festival	Baja	Baja
RF21	Buscar festivales	Baja	Baja
RF22	Insertar evidencias (foto, video o documento) en el festival	Alta	Media
RF23	Eliminar evidencias (foto, video o documento) en el festival	Baja	Baja
RF24	Mostrar evidencias (foto, video o documento) de los festivales	Baja	Media
RF25	Mostrar programa artístico por manifestación	Baja	Baja
RF26	Generar PDF del programa artístico por manifestación	Baja	Media
RF27	Autenticar usuario	Media	Media

2.2.2. Requisitos no funcionales

Entre las características que describen al sistema se encuentran:

- Restricciones de diseño e implementación.

- **RnF1.** El marco de trabajo que se utilizará está basado en el uso de herramientas no privativas:
 - Paradigm v15.1 como herramienta CASE para el modelado. .
 - Microsoft Visual Studio Code v1.68 como IDE de codificación .
 - Python v3.10.5 para la programación del lado del servidor.
 - Django v4.0 como framework de desarrollo web.
 - JavaScript y JQuery 3.6 0 para la programación del lado del cliente.
 - HTML v5 para la estructura del documento web.
 - CSS v3 para estilizar el documento web.
 - Bootstrap v5.1 como biblioteca de estilos.
- **RnF2.** Se utilizará como metodología de software la variación de AUP en su variante UCI.
- Seguridad
 - **RnF3.** El acceso a la información debe estar restringido por usuario, contraseña y rol. El usuario anónimo solo tendrá acceso a las funcionalidades de mostrar y listar información.
 - **RnF4.** Cuando un usuario se autentique en el sistema se le brindará la información correspondiente con su rol.
- Eficiencia
 - **RnF5.** El sistema debe permitir que los usuarios (150) interactúen con él de manera concurrente.
 - **RnF6.** El tiempo de demora de una petición al servidor debe ser menor a cinco (5) segundos
- Usabilidad
 - **RnF7.** El sistema debe poseer una interfaz intuitiva, que permita realizar búsquedas y filtrado.
- Hardware y software requerido para usar la aplicación.
 - **RnF8.** Sistema Operativo: El sitio debe ser accesible desde cualquier sistema operativo.
 - **RnF9.** Al sistema se podrá acceder desde cualquier navegador web.
 - **RnF10.** La computadora que se utilice para montar el servidor debe contar como mínimo con 4 GB de RAM y de almacenamiento 80 GB.

2.3. Descripción de las historias de usuario

Siguiendo lo que plantea la metodología AUP para la UCI, en su escenario cuatro, se generaron un total de 27 historias de usuario para lograr el encapsulamiento de los requisitos, el resto se pueden encontrar en el anexo .3.

A continuación, se se muestran las correspondientes a los requisitos “Crear Unidad artística” y “Listar Festival respectivamente”:

Tabla 2.2. Historia de usuario # 1

Historia de usuario	
Número: 1	Nombre: Crear Unidad artística
Usuario: Encargado facultad, encargado de universidad, encargado provincial y encargado nacional.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
<p>Descripción: El sistema debe permitir crear una unidad artística a través de un formulario donde el usuario rellena todos los campos.</p> <ul style="list-style-type: none"> • Nombre: Campo de texto. Admite caracteres especiales, letras y números. • Género: Campo de selección. • Manifestación: Campo de selección. • Provincia: Campo de selección. • Obra: Campo de texto. Admite caracteres especiales, letras y números. • Autor: Campo de texto. Admite caracteres especiales, letras y números. • Acompañamiento: Campo de texto. Admite caracteres especiales, letras y números. • Duración: Campo de texto. Solo admite números. • Recursos: Campo de texto. Admite caracteres especiales, letras y números. • Miembros: Campo de selección. Se podrá seleccionar más de uno. • Instructor: Campo de selección. Se podrá seleccionar solo uno. • Link: Campo de texto. Solo admite Localizador de Recursos Uniforme (URL, por sus siglas en inglés). 	
<p>Observaciones:</p> <ul style="list-style-type: none"> • El usuario debe estar autenticado con el rol correspondiente. • Si el usuario introduce la información de forma incorrecta, el sistema emite un mensaje notificando el error. • Si el usuario introduce la información dejando campos obligatorios vacíos, el sistema emite un mensaje indicándole que los campos obligatorios deben llenarse. • si es Artes Visuales se permiten 10 miembros, si es Música o Danza se permiten 20 miembros donde hasta el 30 por ciento puede ser egresado. El sistema debe permitir ubicar link de encuesta de satisfacción de la unidad artística. 	

Continúa en la próxima página

Tabla 2.2. Continuación de la página anterior

Interfaz:

Crear Unidad Artística

Nombre

Manifestación

Instructor

Duración

Miembros

Tabla 2.3. Historia de usuario # 2

Historia de usuario																					
Número: 2	Nombre: Listar Festival																				
Usuario: usuario autenticado, usuario anónimo, admin, encargado facultad, encargado de universidad, encargado provincial y encargado nacional.																					
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja																				
Puntos estimados: 1	Iteración asignada: 1																				
Programador responsable: Alejandro Campmany Cejas																					
Descripción: El sistema deberá listar todos los festivales empezando por los mas recientes.																					
Observaciones: No será necesario autenticarse para acceder a dicha funcionalidad. En caso de estar autenticado se mostrará un botón para filtrar por los festivales en los que ha participado dicho usuario.																					
<p>Interfaz:</p> <div style="border: 1px solid gray; padding: 10px; margin: auto; width: 80%;"> <div style="background-color: #003366; color: white; padding: 2px; font-weight: bold; font-size: small;">Festivales de Artistas Aficionados</div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: left; font-size: x-small;"> <thead> <tr style="background-color: #f2f2f2;"> <th style="width: 25%;">Nombre</th> <th style="width: 15%;">Lugar</th> <th style="width: 10%;">Nivel</th> <th style="width: 20%;">Fecha</th> <th style="width: 30%;">Jurado</th> </tr> </thead> <tbody> <tr> <td>Festival Jose Martí</td> <td>Wilfredo Lam</td> <td>Facultad</td> <td>21 de octubre de 2022</td> <td>• Duanny siff 🔍 ✖</td> </tr> <tr> <td>Festival Camilo</td> <td>Wilfredo Lam</td> <td>Universidad</td> <td>24 de octubre de 2022</td> <td>• Duanny siff 🔍 ✖</td> </tr> <tr> <td>Festival Mella</td> <td>Wilfredo Lam</td> <td>Selección Provincial</td> <td>21 de octubre de 2022</td> <td>• Duanny siff 🔍 ✖</td> </tr> </tbody> </table> <p style="text-align: center; font-size: x-small; margin: 5px 0;">Página de .</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <input style="width: 60px;" type="button" value="Mis Festivales"/> <input style="width: 60px;" type="button" value="Crear Festival"/> </div> </div>		Nombre	Lugar	Nivel	Fecha	Jurado	Festival Jose Martí	Wilfredo Lam	Facultad	21 de octubre de 2022	• Duanny siff 🔍 ✖	Festival Camilo	Wilfredo Lam	Universidad	24 de octubre de 2022	• Duanny siff 🔍 ✖	Festival Mella	Wilfredo Lam	Selección Provincial	21 de octubre de 2022	• Duanny siff 🔍 ✖
Nombre	Lugar	Nivel	Fecha	Jurado																	
Festival Jose Martí	Wilfredo Lam	Facultad	21 de octubre de 2022	• Duanny siff 🔍 ✖																	
Festival Camilo	Wilfredo Lam	Universidad	24 de octubre de 2022	• Duanny siff 🔍 ✖																	
Festival Mella	Wilfredo Lam	Selección Provincial	21 de octubre de 2022	• Duanny siff 🔍 ✖																	

2.4. Estilo arquitectónico

Un estilo arquitectónico representa a una familia de sistemas de software que comparten características estructurales fundamentales. Mediante un estilo arquitectónico se puede definir a grandes rasgos la estructura de un sistema de software, mostrando las funciones que tendrán los componentes principales y sus relaciones. Existe alrededor de una docena de estilos arquitectónicos (Maximiliano, 2021).

Para el diseño de la propuesta de solución se utilizará 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

La arquitectura de software brinda soluciones especializadas en algunos dominios, lo cual permite simplificar el proceso de construcción de nuevos sistemas, a través de la reutilización de la infraestructura existente, reduciendo costos y facilitando el mantenimiento de los sistemas. Un patrón arquitectónico establece que son un conjunto de reglas de diseño que identifica los tipos de componentes y conectores que se pueden utilizar para estructurar el sistema (Varela; Sanabria y S. V. Rodríguez, 2021).

El Modelo-Vista-Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo. El Modelo contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia. La Vista o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste. El Controlador actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.

El framework Django está basado en la arquitectura de MVC, pero en Django se realiza una variación dando lugar a una nueva estructura denominada Model-View-Template (MVT, por sus siglas en inglés) (Template: plantilla). El funcionamiento es muy similar con la diferencia de que el controlador pasa a llamarse "vista" y la vista se denomina como "template". A partir de lo planteado anteriormente se asumirá el estilo arquitectónico MVT de Django .

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 (jvatpoint, 2021).



Figura 2.1. Estilo arquitectónico Model-Template-View

2.5. Diagrama de clases de diseño

Un diagrama de clase es un tipo de diagrama Lenguaje Unificado de Modelado (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 se presenta el diagrama correspondiente a la gestión de los festivales.

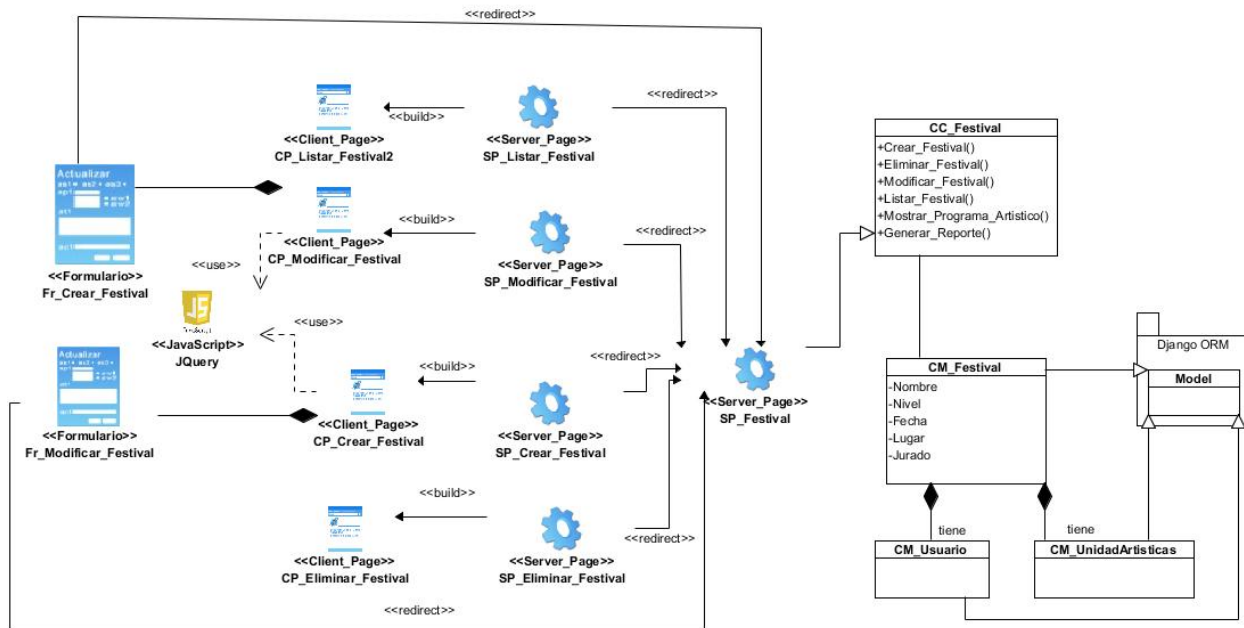


Figura 2.2. Diagrama de clases.(Fuente: Elaboración propia).

Descripción de las clases

- Fr_Crear_Festival: Formulario que contiene la información para la creación de un festival.
- Fr_Modificar_Festival: Formulario que contiene la información para la modificación de un festival.
- CP_Crear_Festival: Página HTML donde se encuentra el formulario para la creación de un festival. Usa JavaScript para realizar validaciones.
- CP_Modificar_Festival: Página HTML donde se encuentra el formulario para la modificación de un festival. Usa JavaScript para realizar validaciones.
- CP_Eliminar_Festival: Página HTML para eliminar festival.
- CP_Listar_Festival: Página HTML donde se listan los festivales
- SP_Crear_Festival: Página del servidor encargada de construir la página cliente Crear festival.
- SP_Modificar_Festival: Página del servidor encargada de construir la página cliente Modificar festival.
- SP_Listar_Festival: Página del servidor encargada de construir la página cliente Listar festival.
- SP_Eliminar_Festival: Página del servidor encargada de construir la página cliente Eliminar festival.
- SP_Festival: Página del servidor encargada de manejar los festivales.
- CC_Festival: Clase Controladora donde se implementan todos los métodos del festival. Se encarga de manipular los modelos.
- CM_Festival: Modelo donde se guardan todos los datos del festival que se almacenan en la base de datos. A su vez son manipulados por el ORM de Django.
- CM_Usuario : Modelo donde se almacena el usuario encargado de gestionar el festival.
- CM_UnidadArtística: Modelo donde se almacenan los datos de las unidades artísticas que participan en un festival.

2.6. Patrones de diseño

Patrones GRASP

El uso de los patrones General Responsibility Assignment Software Patterns (GRASP, por sus siglas en inglés) ayuda al entendimiento de la aplicación y a diseñar software con mayor calidad aplicando buenas prácticas que provienen de la experiencia de años de desarrollo (Ortega, 2021). A continuación se abordan los patrones de este tipo empleados:

- **Experto:** Se usa al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Da origen a diseños donde el objeto de software realiza las operaciones que normalmente se aplican a la cosa real que representa, por lo que ofrece una analogía con el mundo real. Este patrón está presente en las clases controladoras Festival.py, Premio.py y UnidadArtística.py; las cuales tienen implementadas las funcionalidades de crear, obtener, modificar y eliminar de las entidades que manejan respectivamente.

- **Creador:** Este patrón ayuda a identificar quién debe ser el responsable de la creación (o instancia) de nuevos objetos o clases. Se evidencia en las clases Festival.py, Unidad_Artística.py al crear instancias de festivales y unidades artísticas.
- **Bajo acoplamiento:** Las clases que implementan la lógica del negocio y de acceso a datos se encuentran en el modelo, las cuales no tienen asociaciones con las de la vista o el controlador, lo que proporciona que la dependencia en este caso sea baja. Se evidencia al separar las clases del modelo, de las vistas y de los controladores
- **Alta Cohesión:** Este patrón hace referencia a la medida en la que un componente o clase realiza únicamente la tarea para la cual fue diseñada. Se evidencia en las clases clientes y las servidoras, se encargan cada una de realizar una función específica, permitiendo que las mismas no se saturen.
- **Controlador:** El patrón controlador sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Se evidencia en las clases Festival.py, Unidad_Artística.py, entre otras.

Patrones GOF

Los patrones The Gang of Four (GOF, por sus siglas en inglés) constituyen una herramienta fundamental para cualquier programador. Estos son utilizados para resolver problemas de creación de instancias, abstrayendo dichas creaciones (Portelles Liy; Orellana García; Buedo Hidalgo y Canel Gómez, 2022). Estos patrones describen las formas en que diferentes objetos se pueden organizar para trabajar unos con otros. A continuación se abordan los patrones empleados:

- **Decorador:** Este patrón permite añadir dinámicamente funcionalidad a un objeto. Permitiendo crear nuevas clases que hereden de la primera, pero añadiendo nuevas funcionalidades. Este patrón se evidencia en la herencia que existe en las plantillas HTML donde se hereda de MenuPieDP.html para reutilizar el menú lateral y el pie de página que se mantiene en toda la aplicación.
- **Observer (Observador):** Es un patrón de comportamiento, tiene como función fundamental avisarnos cuando se realiza algún cambio sobre el objeto que estamos observando, garantizando la comunicación entre los componentes. Está compuesto por dos parte, el sujeto que es el objeto cuyo estado quiere vigilarse a largo plazo y los objetos observadores, que han de ser informados de cualquier cambio en el sujeto. Este patrón se aplica en el método signal @receiver el cual se encarga del monitoreo cuando se realiza un modificación en la entidad Premio y luego de ser notificado envía una notificación por correo a todos los miembros de la unidad artística con el premio obtenido.

2.7. Modelo de datos

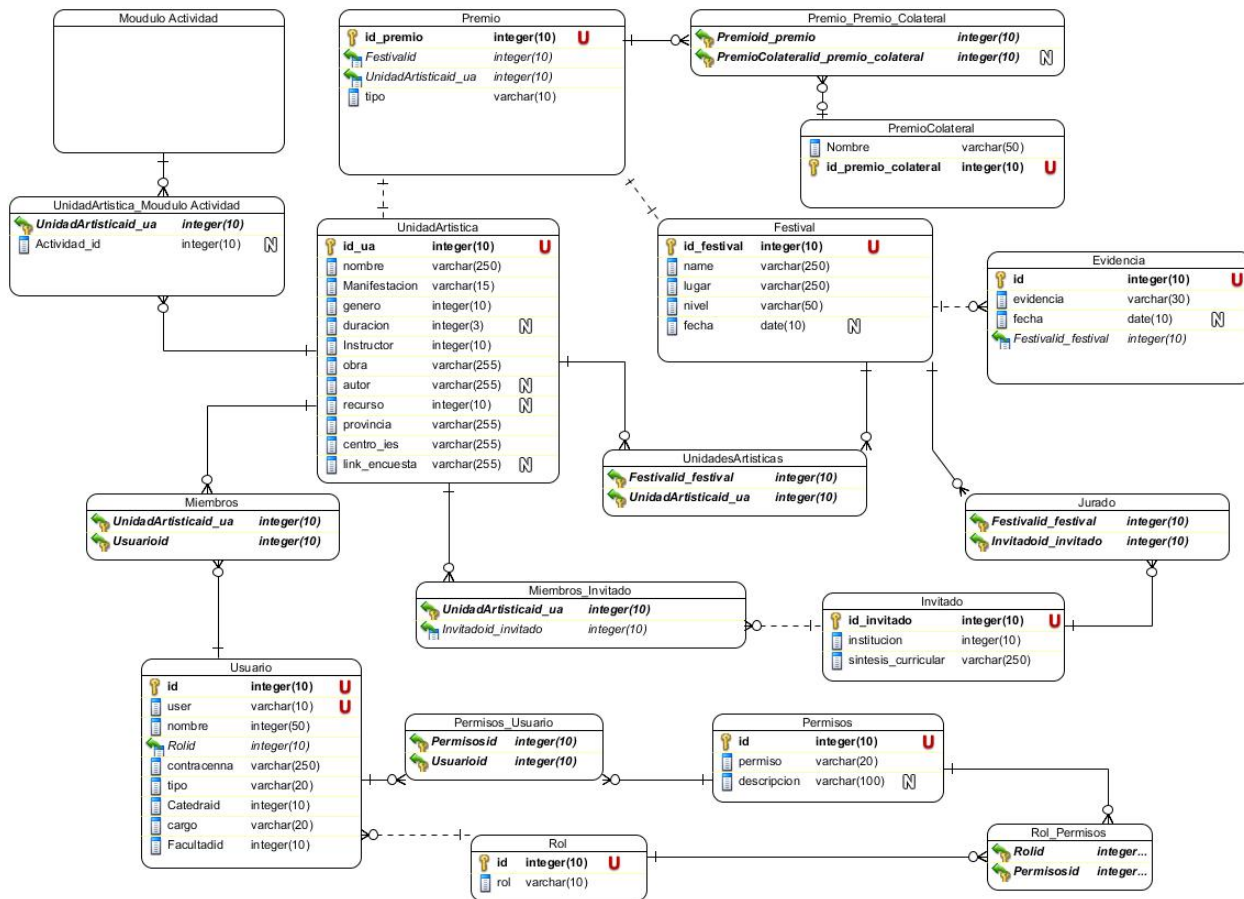


Figura 2.3. Modelo de datos.(Fuente: Elaboración propia).

2.7.1. Descripción de las tablas

1. **Premio**: Tabla principal del modelo de datos en la que se relaciona un festival y una unidad artística donde se asigna un premio o un premio colateral.
2. **Premio_colateral**: Tabla nomencladora que guarda los premios colaterales que existen.
3. **Premio_Premio_colateral**: Tabla producto de la relación de las tablas Premio y Premio_colateral. Donde un premio puede tener asociados varios premios colaterales.
4. **Festival**: Tabla del modelo de datos donde se registran los FAA.
5. **UnidadArtística**: Tabla del modelo de datos donde se registran las unidades artísticas.
6. **UnidadesArtísticas**: Tabla producto de la relación de las tablas Festival y UnidadArtística. Donde se registran las unidades artísticas que participan en un festival.
7. **Jurado**: Tabla producto de la relación de las tablas Festival e Invitado. En esta tabla se registran usuarios de tipo invitado que actúan como jurado en un festival.

8. **Miembros:** Tabla producto de la relación de las tablas Unidad_Artística y Usuario. En esta tabla se registran los miembros de una unidad artística.
9. **Miembros_Invitado:** Tabla producto de la relación de las tablas Unidad_Artística y Invitado. En esta tabla se registran los integrantes de una unidad artística que no forman de la FEU.
10. **Invitado:** Tabla donde se registran las personas que no pertenecen a la FEU.
11. **Modulo Actividad:** Módulo que consume el sistema donde están todas las actividades que se realizan en la comunidad universitaria.
12. **UnidadArtística_Modulo Actividad:** Tabla producto de la relación existente entre las unidades artísticas y las actividades donde se registra la participación de una unidad artística en una actividad.
13. **Evidencias:** Tabla que recoge todas las evidencias de los festivales, donde un festival puede tener muchas evidencias, pero una evidencia pertenece a un único festival.
14. **Usuario:** Tabla que guarda los usuarios del sistema y permite asignarle roles, permisos, así como la autenticación en el sistema. Existen 2 tipos de usuarios del sistema: los trabajadores y los estudiantes. Tiene relación con la tabla facultad lo que permite determinar a qué facultad pertenece, en caso de no pertenecer a ninguna se le asigna la tupla fija de id = 1 y nombre_facultad = "No procede".
15. **Permisos:** Tabla que recoge los permisos que existen con su descripción.
16. **Rol_Permisos:** Tabla producto a la relación de roles con los permisos, donde a cada rol se le hace corresponder varios permisos y a cada permiso le corresponde varios roles.
17. **Permisos_Usuario:** Tabla que relaciona a los permisos con los usuarios directamente, donde a cada usuario le corresponde muchos permisos y a cada permiso le corresponde varios usuarios.
18. **Rol:** Tabla que registra el rol de cada usuario. .

2.8. Conclusiones del capítulo

Luego de haber realizado el análisis y diseño del sistema propuesto y generar los artefactos que dispone la metodología AUP-UCI en el escenario 4, se arriba a las conclusiones siguientes:

- La especificación de los requisitos del sistema permitió describir cada uno de los requisitos funcionales a través de las historias de usuario, así como identificar los requisitos no funcionales de dicho sistema teniendo en cuenta un conjunto de características generales y restricciones del mismo.
- El diseño del modelo de datos permitió definir la estructura de los elementos persistentes en la base de datos, las relaciones entre estos y sus atributos.
- La elaboración de las historias de usuario permitió esclarecer el proceso de desarrollo de la aplicación.
- Al usar el framework de desarrollo Django se utilizará la arquitectura MVC, debido a que es la arquitectura que implementa. Sirvió para elaborar el diseño de clases.
- La utilización de patrones permitirá utilizar buenas prácticas en la implementación y facilita la reutilización de código.

Implementación y prueba del sistema para la gestión del Movimiento de Artistas Aficionados de la UCI

En el presente capítulo se abordan los principales 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. La verificación del correcto funcionamiento del software mediante la aplicación de las pruebas correspondientes y se presenta el nivel de satisfacción de los usuarios de la solución implementada aplicando la técnica de Iadov.

3.1. Estándares de codificación

Entendemos como estándar de código a un conjunto de convenciones establecidas de antemano (denominaciones, formatos, etc.) para la escritura de código. Estos estándares varían dependiendo del lenguaje de programación elegido y además varían en cobertura, algunos son más extensos que otros. No se requiere un código estándar específico, sin embargo, se debe ser consistentes con el método de escritura del código (Patilla; Enciso; Pulache; J. L. L. Rodríguez; Huallanca y Conislla, 2021). A continuación, se describe el estándar de codificación empleado en el Sistema de Gestión del MAA.

- Clases: El nombre de la clase debe estar en concordancia con el nombre del archivo y debe declararse en notación PascalCase.
- Funciones y métodos: Los nombres de funciones deben empezar siempre con una letra minúscula, utilizando el formato camelCase.
- Variables: Los nombres de variables pueden contener caracteres alfanuméricos. El formato definido para los nombres de variables debe ser camelCase.
- Sentencias de Control: Las sentencias de control basadas en las construcciones if y elseif deben tener un solo espacio en blanco antes del paréntesis de apertura del condicional y un solo espacio en blanco después del paréntesis de cierre. La llave de apertura “{” se escribe en la misma línea que la sentencia condicional. La llave de cierre “}” se escribe siempre en su propia línea.

- Declaración de clases: Debe escribirse “ : ” siempre a continuación del nombre de la clase y el contenido de la misma siempre va escrito con margen superior al primer carácter de la declaración.
- Declaración de funciones y métodos: comienzan por la palabra reservada “def”. La llave “{” debe escribirse siempre seguido del nombre. Debe escribirse “ : ” siempre a continuación del nombre de la clase y el contenido de la misma siempre va escrito con margen superior al primer carácter de la declaración.
- Operadores: Se debe colocar un espacio antes y después de los operadores binarios (+, -, +=, !=, ==, etc.).
- Comentarios: Los comentarios deben ser añadidos de forma que resulten prácticos, para explicar el flujo del código y el propósito de las funciones y variables. Utilizar los caracteres “#”.

3.2. Diagrama de despliegue

El Diagrama de Despliegue es un tipo de diagrama del UML. Mediante el diagrama de despliegue se muestra la configuración de los nodos que participan en la ejecución y los componentes que residen en ellos. Muestra la configuración de los elementos de procesamiento en tiempo de ejecución, los componentes de software, hardware, procesos y objetos que los ejecutan (Alvarez Puello y Ortega Camargo, 2020).

- Permiten modelar la disposición física o topología de un sistema.
- Muestra el hardware usado y los componentes instalados en el hardware.
- Muestra las conexiones físicas entre el hardware y las relaciones entre componentes.

A continuación, se presenta el diagrama de despliegue de la solución, donde se muestran componentes a utilizar para desplegar el sistema en cuestión:

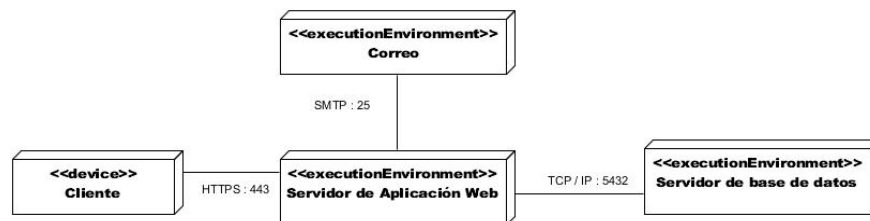


Figura 3.1. Diagrama de Despliegue (fuente: Elaboración propia)

Descripción de los componentes del diagrama:

- Dispositivo Cliente: Se refiere a los dispositivos utilizados por el cliente para acceder a la aplicación vía Protocolo Seguro de Transferencia de Hipertexto (HTTPS, por sus siglas en inglés) por el puerto 443.
- Servidor de aplicaciones: Se encarga de garantizar el acceso a la aplicación y manejar las interfaces. Se comunica con la base de datos mediante el protocolo TCP/IP mediante el puerto 5432.

- Servidor de base de datos: Es el encargado de almacenar y brindar acceso a todos los datos almacenados en el sistema.
- Servidor de correo: Es el encargado de ejecutar las funcionalidades relacionadas con el correo, se comunica con el servidor utilizando el protocolo Protocolo simple de transferencia de correo (SMTP, por sus siglas en inglés) por el puerto 25.

3.3. Pruebas de software

Las pruebas de software sirven para detectar los defectos del sistema y reducir los riesgos asociados. Permiten el aseguramiento de la calidad para lograr la satisfacción del cliente y tiene un impacto directo en el costo y el desarrollo del producto. Garantizando que el software y los procesos en el ciclo de vida se ajustan a las necesidades específicas (Serna; Martínez y Tamayo, 2019).

3.3.1. Estrategia de pruebas

Tabla 3.1. Estrategia de pruebas. (Fuente: Elaboración propia).

Pruebas	Método	Herramienta	Alcance
Funcional	Caja negra		Se probará el funcionamiento correcto de todos los requisitos.
Unitaria	Caja blanca con la técnica del camino básico		Se automatizarán pruebas para las unidades de código separadas por módulos.
Rendimiento	Pruebas de carga y estrés	Apache JMeter v5.5	Se aplicará sobre un entorno de pruebas con prestaciones menores que el de despliegue, debido a falta de recursos. Se probará la aplicación con 150 usuarios concurrentes buscando tiempos de respuesta menores a 5 segundos.
Seguridad		Acunetix web vulnerability scanner 9.5	Se aplicará para detectar vulnerabilidades de seguridad en el sistema.
Continúa en la siguiente página			

Pruebas	Método	Herramienta	Alcance
Aceptación	Pruebas de alpha y beta		<p>Se aplicará en un entorno controlado con el equipo de desarrollo y en la Dirección de Extensión donde se desplegará el sistema. El éxito de la prueba se medirá:</p> <ul style="list-style-type: none"> • Exitosa: más del 75 % de las funcionalidades correctas • Fracaso: menos del 75 % de las funcionalidades correctas
Ejecutado por:	Alejandro Campmany Cejas		

3.3.2. Pruebas unitarias

Las pruebas unitarias son una forma de probar pequeñas e individuales porciones de código. A través de ellas se puede verificar que determinado módulo o funcionalidad se ejecuta dentro de los parámetros y especificaciones concretadas en los requisitos. El objetivo principal de una prueba unitaria es comprobar el correcto funcionamiento de una unidad de código. El método utilizado para realizar este tipo de prueba se denomina caja blanca (Pressman, 2002).

Método de caja blanca

El método de caja blanca tiene como objetivo principal probar el código. Entre las técnicas existentes de caja blanca se encuentran: prueba del camino básico, prueba de condición, prueba de flujo de datos y prueba de bucles. Para el presente trabajo se utiliza la prueba del camino básico, la misma permite: generar el grafo de flujo, calcular la complejidad ciclomática, determinar los caminos linealmente independientes y diseñar los casos de prueba para forzar la ejecución de cada camino del conjunto básico.

La técnica del camino básico fue aplicada al método Crear unidad artística, que se encarga de registrar una nueva unidad artística en el sistema.

Tabla 3.2. Cálculo de la complejidad ciclomática del método Crear unidad artística

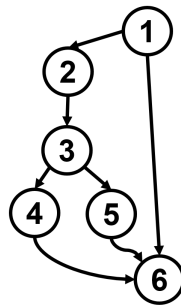
Método

```

32     def post(self,request):
33         if request.method == "POST":
34             form = UnidadArtisticaFrom(request.POST)
35             if form.is_valid():
36                 form.save()
37                 return redirect('unidad-artistica')
38         else:
39             return render(request, self.template_name, {"form": form, "error": form.errors})

```

Grafo resultante:



Complejidad ciclomática:

$V(G) = \# \text{ de regiones}$
 $V(G) = 3$

$V(G) = A - N + 2$
 $V(G) = 7 - 6 + 2$
 $V(G) = 3$

$V(G) = P + 1$
 $V(G) = 2 + 1$
 $V(G) = 3$

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 obtuvo que la complejidad ciclomática es igual a 3, lo que significa que existen tres posibles caminos linealmente independientes y hay que diseñar un mínimo de tres casos de prueba para el algoritmo Crear unidad artística. La tabla 3.3 muestra los caminos existentes.

Tabla 3.3. Caminos del grafo de flujo (Fuente: Elaboración propia).

No.	Camino
1	1,2,3,4,6
2	1,2,3,5,6
3	1,2,6

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

Proceso:
Crear unidad artística.
Casos de prueba:
Crear unidad artística.. Escenario 1.1.
Camino independiente:
1,2,3,4,6
Entradas:
<ul style="list-style-type: none"> ● Nombre: Festival UCI 2022. ● Lugar: Wilfredo Lam ● Fecha: 22/12/2022 ● Nivel: Universidad ● Jurado: Juan Pérez Pérez, Pedro González Rodríguez ● Unidades Artísticas: Grupo Espacio Abierto, Elementrix.
Resultados esperados:
<ul style="list-style-type: none"> ● Adicionar una nueva unidad artística. ● Mostrar el mensaje “Se ha registrado la unidad artística”.
Condiciones de ejecución:
<ul style="list-style-type: none"> ● Usuario autenticado. ● Usuario con rol de encargado.

Con la realización de los casos de prueba diseñados se probó la ejecución de cada sentencia del código al menos una vez, teniendo en cuenta todas las condiciones lógicas en sus variantes verdaderas y falsas. Los resultados del método de caja blanca fueron satisfactorios. La obtención de la complejidad ciclomática de valor tres permitió determinar que el algoritmo es de baja complejidad.

3.3.3. Pruebas funcionales

Las pruebas funcionales son diseñadas tomando como referencia las especificaciones funcionales de un componente o sistema (lo que vamos a probar, el software o una parte de él). Se realizan para comprobar si el software cumple las funciones esperadas. Las pruebas funcionales se basan en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para el software. El objetivo principal de las pruebas funcionales es analizar el producto terminado y determinar si hace todo lo que debería hacer y

si lo hace correctamente. El método utilizado para llevar a cabo este tipo de prueba es caja negra (Pressman, 2002)

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 (ibíd.). El método de caja negra presenta varias técnicas de prueba como son: partición de equivalencia, análisis de valores límites y grafos de causa-efecto.

En la presente investigación se utilizará 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.

A continuación, la tabla 3.5 muestra el diseño de caso de pruebas del requisito “Crear festival” donde se analizarán las variables y condiciones que puedan determinar la respuesta del sistema.

Tabla 3.5. Caso de prueba “Crear festival” (Fuente: Elaboración Propia).

Escenario	Descripción	Variables						Respuesta esperada	Respuesta
		Nombre	Duración	Lugar	Unidades artísticas	Jurado	Nivel		
EC 1.1. Crear festival correctamente	El usuario debe introducir los datos de los campos mostrados y dar clic a en botón “Aceptar”	V Festival de la Facultad 1 2022	V 9	V Plaza Me-lla	V Ele-men-trix, Grupo Es-pacio Abier-to	V Juan Pérez Pérez	V Fa-cultad	Se accede al sistema, se llenan correctamente los campos del festival y se muestra un mensaje de confirmación de la creación del festival.	El sistema crea el festival correctamente y muestra un mensaje de confirmación

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA PARA LA GESTIÓN DEL MOVIMIENTO DE ARTISTAS AFICIONADOS DE LA UCI

EC 1.2.	Crear festival con campos obligatorios vacíos	El usuario debe introducir los datos de los campos mostrados y dar clic a en botón “Entrar”	V Festival de la Facultad 1 2022	V 8	V Plaza Me-lla	V	V	I	Se accede al sistema, se deja el campo obligatorio nivel vacío y el sistema debe mostrar un mensaje de error donde muestre que se debe rellenar dicho campo.	El sistema muestra el mensaje de error “Rellene el campo nivel”.
EC 1.3.	Crear festival con campos incorrectos	El usuario debe introducir los datos de los campos mostrados y dar clic a en botón “Aceptar”	V Festival de la Facultad 1 2022	I cinco	V Plaza Me-lla	V Ele-mentrix, Grupo Es-pacio Abier-to	V Juan Pérez Pérez	V Fa-cultad	Se accede al sistema, se re-llena mal el campo fecha y el sistema debe mostrar un mensaje de error que in-dique que el campo fecha es incorrecto.	El sistema muestra el mensaje de error “Campo duración incorrecto”

Tabla 3.6. Variables de caso de prueba “Crear festival”(Fuente: Elaboración Propia).

No.	Variable	Valor Nulo	Descripción
1	Nombre	No	Es un campo que permite al usuario insertar el nombre del festival. Acepta letras, números y espacio entre letras.
2	Lugar	No	Es un campo que permite al usuario insertar el nombre del lugar donde se realizará el festival. Acepta letras, números y espacio entre letras.
3	Fecha	No	Es un campo que permite al usuario insertar la fecha de realización del festival. Acepta solo datos de tipo fecha de la forma 28/12/2023.
4	Nivel	No	Es un campo de selección que permite seleccionar el nivel del festival.
5	Jurado	Si	Es un campo que permite seleccionar los usuarios de tipo invitado que realizan función de jurado.
6	Unidades Artísticas	Si	Es un campo que permite seleccionar las unidades artísticas que participarán en el festival.

Las pruebas de caja negra se aplicaron con el objetivo de evaluar las interfaces de comunicación con el usuario, las que demostraron coherencia y funcionalidad. 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 las no conformidades arrojadas y perfeccionando lo obtenido. Este tipo de pruebas se aplica de conjunto con otras, como por ejemplo las pruebas de aceptación.

Resultados de las pruebas aplicadas

Durante la realización de las pruebas se detectaron un conjunto de no conformidades relacionadas con errores de validación y funcionalidad. Los resultados se muestran en la figura 3.2, donde se evidencia la cantidad de casos de prueba ejecutados, los casos de prueba con no conformidades y las no conformidades identificadas, las cuales fueron corregidas.

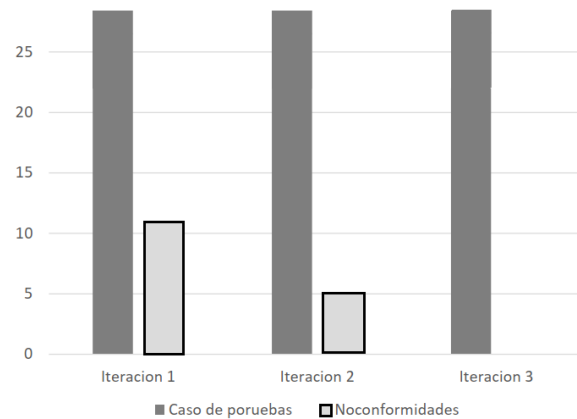


Figura 3.2. Resultados de Pruebas (Elaboración Propia)

Se realizaron tres iteraciones, durante la primera iteración se analizaron 27 casos de prueba, de los cuales cuatro resultaron tener un total de 11 no conformidades. En la segunda iteración a través de las pruebas de regresión se verificó que las no conformidades anteriores estuviesen solucionadas sobre los casos de prueba que incidían; de estas pruebas se obtuvo 5 no conformidad, quedando resuelta en la tercera iteración y cumpliéndose correctamente los requerimientos del sistema.

3.3.4. Pruebas de seguridad

Es un tipo de prueba de software que revela vulnerabilidades, amenazas, riesgos en una aplicación de software y previene ataques maliciosos de intrusos. El propósito de las Pruebas de Seguridad es identificar todas las lagunas y debilidades potenciales en el sistema de software que podrían resultar en la pérdida de información. Se realizan con el objetivo de detectar posibles riesgos de seguridad en el sistema, identificar las amenazas y medir las vulnerabilidades potenciales para ser corregidas por los desarrolladores posteriormente. Las pruebas de seguridad se aplicaron utilizando la herramienta Acunetix Web Vulnerability Scanner 9.5 la cual realiza alertas por tipo: alta, media, baja e informativa. Se realizó el escáner mediante dos iteraciones con dicha herramienta. Durante la primera iteración se encontraron 86 alertas de seguridad, de ellas 4 fueron de nivel medio, 5 fueron de nivel bajo y 77 fueron informativas como muestra la figura 3.3.

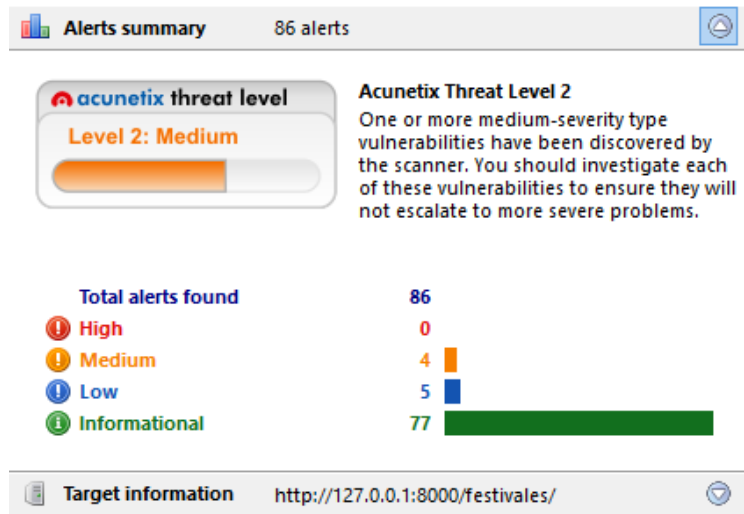


Figura 3.3. Resultados de las pruebas de seguridad, primera iteración

A continuación se describen las principales vulnerabilidades encontradas en el sistema:

- Las credenciales de usuario se transmiten a través de un canal no cifrado.
- Existe un formulario con un campo de contraseña el cual envía datos de usuario utilizando el método GET, por lo tanto, el contenido del campo de contraseña aparecerá en la URL.
- Se encontró un formulario HTML aparentemente sin protección implementada contra la técnica llamada Falsificación de petición en sitios cruzados (CSFR, por sus siglas en inglés).
- Existen enlaces rotos que no dirigen a ningún lugar y terminan en un error, aunque no son inaccesibles.

Para dar solución a dichos problemas se implementó la protección CSFR en los formularios, se eliminaron los enlaces rotos que aparecían en el código, se utilizaron canales cifrados para la transmisión de credenciales de usuario y se actualizaron correctamente los métodos GET y POST para evitar evitar que se mostraran datos en la URL. Luego de realizar una segunda iteración se demostró que las vulnerabilidades fueron resueltas como se muestra en la figura 3.4. Donde solo aparecieron dos alertas informativas que no afectan la seguridad de la aplicación.

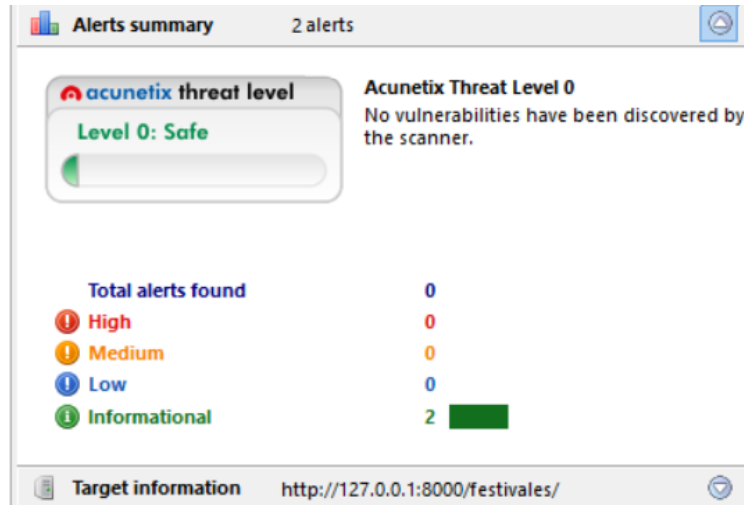


Figura 3.4. Resultados de las pruebas de seguridad, segunda iteración

3.3.5. Pruebas de rendimiento

Las pruebas de carga y estrés son realizadas con el objetivo de comprobar el funcionamiento del sistema bajo condiciones planificadas. Estas pruebas demuestran si la aplicación cumple con los diferentes criterios de rendimiento, permitiendo conocer con precisión las estadísticas referidas a los diferentes procesos que se ejecuten durante el uso del sistema. Para realizar las pruebas se utiliza la herramienta JMeter, la cual es una aplicación multiplataforma que permite simular el uso de diferentes aplicaciones, creando registros de rendimiento de ellas para facilitar su análisis Milián; J. F. R. Pérez y Morejón, 2021. Para la realización de la prueba se tomaron 300 usuarios para simular el comportamiento real del sistema.

El entorno elegido para las pruebas posee las siguientes características:

- **Microprocesador:** Intel(R) Core(TM) i5-2520M CPU 2.50GHz.
- **Memoria RAM:** 4 GB.
- **Tarjeta de red:** Ethernet 10/100 Mbps.
- **Sistema operativo:** Windows 8.1, Arquitectura 64 bits.

Para la realización de las pruebas de carga se seleccionó una muestra de 150 usuarios concurrentes. Al realizar dicha prueba se obtuvo un tiempo de respuesta promedio del servidor inferior a los 5 segundos como se muestra en la figura 3.5 establecidos en el requisito no funcional

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt...
Petición HTTP	300	10.054	740	18110	7314.74	0.00%	3.0/sec	114.17	5.05	10.465.0
Total	300	10.054	740	18110	7314.74	0.00%	3.0/sec	114.17	5.05	10.465.0

Figura 3.5. Resultados de la prueba de carga

Para la prueba de estrés se tomó una muestra de 250 usuarios concurrentes como muestra la figura 3.6 dando un tiempo de respuesta superior los 5 segundos como se muestra en la figura 3.6.

Etiqueta	# Muestras	Media	Min	Máx	Dev. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt...
Peticion HTTP	300	8154	468	15374	5580,84	30,40%	7,7/sec	83,17	3,05	81055,0
Total	300	8154	468	15374	5580,84	30,40%	7,7/sec	83,17	3,05	81055,0

Figura 3.6. Resultados de la prueba de carga

3.3.6. Pruebas 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 errores y resolverlos (Pressman, 2002), 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: Se realiza en el sitio final donde se va a desplegar el software. 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)
- MsC. Serguey González Garay (Vicedecano de Extensión)
- Ing. Álvaro Alejandro Acosta Ruiz (Especialista B en Ciencias Informáticas)
- Lic. César Cuten Díaz (Profesor de teatro)
- Lic. Rosa María Hernández (Profesora de literatura)
- Ing. Marta Acosta Álvarez (Profesora de literatura)
- Lic. Siry Frades Guzmán (Profesora de música)
- Lic. Carmen Tamara Echeverría (Profesora de música)
- Ing. Julio Jesús García Coste (Profesor de danza)
- Ing. Adis Mirtha Canet Aliaga (Especialista B en Ciencias Informáticas)

Luego de haber aplicado las pruebas de aceptación tanto en un entorno controlado (equipo de desarrollo) y en un entorno libre (cliente) se fueron arrojando no conformidades que fueron resueltas. Dentro de las no conformidades arrojadas por las pruebas Alfa se encontraron errores en funcionalidades que no cumplían con las especificaciones del cliente y las arrojadas por las pruebas Beta se pueden resumir en errores en el proceso de creación de unidades artísticas y el envío de notificaciones.

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, ya que 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 9) (Gallegos; Valenzuela; M. G. 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 3.7

Tabla 3.7. Cuadro Lógico de Iadov (Fuente: Elaboración propia).

	1- ¿Considera usted que es posible participar en el festival de artistas aficionados sin la entrega de la documentación establecida?								
	NO			NO SÉ			SÍ		
3- ¿Satisface sus necesidades, según el rol que desempeña en la gestión del movimiento de artistas aficionados, el sistema propuesto?	2- ¿Utilizaría usted la aplicación propuesta a la hora de generar la documentación referente al festival de artistas aficionados?								
	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 tomó una muestra de 10 personas. De estos, 2 ostentan la categoría de Máster en ciencias, 4 Ingenieros y 4 Licenciados, todos con un amplio conocimiento en el proceso de extensión y el 80 % de ellos han impartido cursos de formación y capacitación en la Dirección de Extensión de la UCI. La población seleccionada fueron los mismos que se le aplicaron las pruebas de aceptación.

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 0.95 lo que significa una clara satisfacción con el uso del sistema.

$$ISG = \frac{9(+1) + 1(+0,5) + 0(0) + 0(-0,5) + 0(-1)}{10} = 0.95$$

A partir de la operacionalización de las variables definidas en la introducción de la investigación y luego de la aplicación de la técnica de Iadov que permite conocer el nivel de satisfacción de los usuarios, se puede llegar a la conclusión de que el desarrollo de un sistema informático para la Dirección de Extensión Universitaria debe contribuir a la gestión de información del Movimiento de Artistas Aficionados en la Universidad de las Ciencias Informáticas.

3.5. Consideraciones parciales

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

- La implementación de la propuesta de solución facilitó obtener una aplicación funcional capaz de contribuir con la gestión de información del MAA en 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 realización de las pruebas de seguridad permitió identificar y corregir vulnerabilidades en el sistema.
- La realización de las pruebas de rendimiento sirvió para optimizar el código y garantizar que el sistema se comporte adecuadamente cuando sea empleado por varios usuarios al mismo tiempo.
- La aplicación de la técnica de Iadov 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 como una clara satisfacción con el uso del sistema informático propuesto.

Con el desarrollo de la presente investigación se obtuvo un Sistema informático que contribuirá con la gestión de la información del Movimiento de Artistas Aficionados. Permitiendo que la información se encuentre accesible, ordenada y centralizada. Los objetivos propuestos fueron cumplidos con satisfacción, por lo que se puede arribar a las siguientes conclusiones:

- Con el análisis de sistemas homólogos se pudo identificar tendencias y herramientas utilizadas en el desarrollo de sistemas informáticos para la gestión de información.
- La selección de herramientas y tecnologías facilitó guiar el desarrollo del sistema.
- Con la implementación del Sistema informático para la gestión del Movimiento de Artistas Aficionados en la Universidad de las Ciencias Informáticas se obtuvo una herramienta que dió solución a la problemática planteada inicialmente por el cliente.
- La definición y especificación de la arquitectura, requisitos funcionales, no funcionales y los artefactos generados por la metodología, permitieron obtener las capacidades del sistema y estructurar la solución implementada.
- Con el desarrollo de las pruebas se pudo detectar y corregir errores cometidos durante la fase de implementación, garantizando el cumplimiento de las especificaciones requeridas.
- Con la validación de la solución propuesta mediante las pruebas de software internas y de aceptación, además de la valoración de la satisfacción con el uso del sistema mediante la técnica y de Iadov se validó el correcto funcionamiento del sistema, el cumplimiento de las necesidades plasmadas con la adecuada calidad y su aptitud para ser usado.

Recomendaciones

Para dar continuidad a la presente investigación se recomienda:

- Integrar la aplicación con el Sistema de Información Gerencial para la Extensión Univerrsitaria.

Referencias bibliográficas

- ALVAREZ PUELLO, Jenniffer Helena y ORTEGA CAMARGO, Amaury Rafael, 2020. Plataforma tecnológica para asistir la etapa de construcción y despliegue de código fuente en proyectos de la asignatura ingeniería de software (vid. pág. 30).
- 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. 9).
- BATISTA, Amado, 2016. Estrategia metodológica de integración de procesos sustantivos universitarios, contribución de la extensión universitaria a la promoción de salud en la Universidad de La Habana. *La Habana: Universidad de La Habana* (vid. pág. 6).
- CODE, Visual Studio, 2022. *Visual Studio Code*. Url: <https://code.visualstudio.com/> (vid. pág. 15).
- COLOMBIA, 2016. *Sistema para la prevención de la deserción de la educación superior*. Autor Bogotá (vid. pág. 10).
- DE ARMA-HERNÁNDEZ, Arianna y SABLÓN-FERNÁNDEZ, Luis E, 2019. Aplicación web para la gestión de la información especializada en Geociencia. *Ciencia & Futuro*. Vol. 9, n.º 2, págs. 106-127 (vid. pág. 13).
- DEFINICIÓN.DE, 2022. *Proceso*. Url: <https://definicion.de/?s=proceso> (vid. pág. 7).
- FOUNDATION, OpenJS, 2022. *¿Qué es jQuery?* Url: <https://jquery.com/> (vid. pág. 15).
- 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 extra-escolar 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. 42).
- GETBOOTSTRAP, 2022. *getbootstrap*. Url: <https://getbootstrap.esdocu.com/> (vid. pág. 14).
- GONZÁLEZ APORTELA, Odette; GONZÁLEZ FERNÁNDEZ-LARREA, Mercedes; ZAMBRANO LOOR, Tania Miladi; BALSINDE HERRERA, José de la Caridad y BATISTA MAINEGRA, Amado, 2022. Retos en la gestión del proceso extensionista cubano: sistema de información gerencial. *Mendive. Revista de Educación*. Vol. 20, n.º 4, págs. 1146-1159 (vid. pág. 8).

- GONZÁLEZ, González; RAMÓN, Gil y MERCEDES, González Fdez-Larrea, 2001. Programa nacional de extensión universitaria. *La Habana. Cuba*. Vol. 230 (vid. págs. 1, 7).
- JAVATPOINT, 2021. *Django MVT*. Url: <https://www.javatpoint.com/django-mvt> (vid. pág. 23).
- JIMÉNEZ, Javier Guillot. SIGUA: SISTEMA INFORMÁTICO DE GESTIÓN UNIVERSITARIA (vid. pág. 11).
- JOVENCLUB, Revista, 2020. *Gestion de informacii; /zn*. Url: <https://revista.jovenclub.cu/gestion-de-informacion/> (vid. pág. 9).
- LANGUAGES, Oxford, 2022. Url: <https://languages.oup.com/google-dictionary-es> (vid. pág. 7).
- LOZANO, MsC. Fernando Picayo, 2018. *Comunidad sigenu*. Url: <http://sigenu-comunidad.cujae.edu.cu:8081/sigenu-comunidad> (vid. pág. 12).
- MARTÍNEZ CABRERA, Lídice, 2019. *Sistema para la Gestión de la Información Estadística del Festival de Artistas Aficionados*. B.S. thesis. Universidad de las Ciencias Informáticas. Facultad 4 (vid. págs. 1, 8).
- MAXIMILIANO, Cristiá, 2021. Una Teoría para el Diseño de Software (vid. pág. 23).
- MES, 2022. *Movimiento de Artistas Aficionados*. Url: <https://www.mes.gob.cu/movimiento-de-artistas-aficionados> (vid. págs. 2, 9).
- MILIÁN, Jorge Luis Peña; PÉREZ, José Felipe Ramírez y MOREJÓN, Madelayne Muñoz, 2021. Módulo Cartas Avales para el Sistema Informático Colpadi de la Unidad Central de Cooperación Médica. Cuba. *Revista Cubana de Informática Médica*. Vol. 13, n.º 1 (vid. pág. 40).
- MOZILLA, Developer, 2022. *Developer Mozilla*. Url: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics (vid. pág. 14).
- MOZILLA.ORG, 2022. *Fundamentos de JavaScript*. Url: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics (vid. pág. 14).
- ORTEGA, Gilberto Andrés Vargas, 2021. Lineamientos para el diseño de aplicaciones web soportados en patrones GRASP. *Ciencia e Ingeniería*. Vol. 8, n.º 2, págs. e5716304-e5716304 (vid. pág. 25).
- PACHECO SUREZ Y., Infante Jaime, 2012. Sistema automatizado para la gestión de la extensión universitaria. *Universidad de Pinar del Río* (vid. pág. 11).
- PATILLA, Hubner Janampa; ENCISO, Edgar Gómez; PULACHE, José Carlos Juárez; RODRÍGUEZ, Jorge Luis Lozano; HUALLANCA, Eder Solórzano y CONISLLA, Yudith Meneses, 2021. Modelo de Gestión de Desarrollo de Software Ágil mediante Scrum y Kanban sobre la Programación Extrema. *Revista Ibérica de Sistemas e Tecnologias de Informação*. N.º E43, págs. 450-466 (vid. pág. 29).

- 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. 9).
- PORTELLES LIY, Adrian; ORELLANA GARCÍA, Arturo; BUEDO HIDALGO, Denys y CANEL GÓMEZ, Dairon, 2022. Funcionalidades de estadísticas y reportes para el Repositorio cubano de datos de neurociencias. *Revista Cubana de Informática Médica*. Vol. 14, n.º 1 (vid. pág. 26).
- POSTGRESQL, 2022. *PostgreSQL*. Url: <https://www.postgresql.org/> (vid. pág. 15).
- PRESSMAN, Roger S, 2002. *Ingeniería de Software: Un enfoque práctico*. McGraw Hill New York (vid. págs. 13, 32, 35, 41).
- PROJECT, Django, 2022. *Django Project*. Url: <https://www.djangoproject.com/> (vid. pág. 14).
- 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. 11).
- RODRÍGUEZ, Marbelis Rojas; GAMBOA, Yannia Moreira y LÓPEZ, Surayne Torres, 2019. La Integración Continua pilar fundamental en el proceso de desarrollo de software. *Serie Científica de la Universidad de las Ciencias Informáticas*. Vol. 12, n.º 10, págs. 102-116 (vid. pág. 15).
- SERNA, Edgar; MARTÍNEZ, Raquel y TAMAYO, Paula, 2019. Una revisión a la realidad de la automatización de las pruebas del software. *Computación y Sistemas*. Vol. 23, n.º 1, págs. 169-183 (vid. pág. 31).
- SNIES, 2012. *Sistemas información*. Url: <https://www.mineduacion.gov.co/sistemasdeinformacion/1735/w3-propertyname-2672.html> (vid. pág. 10).
- TEAM, Microsoft 365, 2019. *La guía sencilla para la diagramación de UML y el modelado de la base de datos*. Url: <https://www.microsoft.com/es-ww/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling#:~:text=El%20Lenguaje%20Unificado%20de%20Modelado,de%20un%20sistema%20o%20proceso.> (vid. pág. 13).
- UCI. Url: <https://www.uci.cu/vida-universitaria/cultura> (vid. pág. 2).
- VARELA, Oscar Agudelo; SANABRIA, Fernando Riveros y RODRÍGUEZ, Santiago Valbuena, 2021. Evaluación de una Arquitectura de Software. *Prospectiva*. Vol. 19, n.º 2, págs. 10 (vid. pág. 23).
- 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. 14).

Apéndices

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

Tabla 8. Guía de observación para estudio de soluciones (Fuente: Elaboración propia).

No.	Indicadores	D	R	B	MB	E
1	¿Son herramientas web?					
2	Apoyo a la toma de decisiones					
3	¿Interfaz gráfica?					
4	¿Presentan permisos de usuario?					
5	¿Gestionan el MAA?					
6	¿Busca información?					

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

- SNIES (1)
- Sistema para la Prevención de la Deserción en la Educación Superior (2)
- Sistema automatizado para la gestión de la Extensión Universitaria (3)
- SIGUA (4)
- Página web extensión universitaria de la Universidad de Holguín (5)
- SIGENU

.2. Modelo de encuesta aplicada

Tabla 9. Encuesta.

<p>Estimado(a):</p> <p>Lea con detenimiento cada una de las preguntas antes de responder. Te agradecemos tu participación y franqueza al decirnos honestamente lo que piensa sobre el uso del sistema para la gestión del MAA.</p> <p>1- ¿Considera usted que es posible participar en el festival de artistas aficionados sin la entrega de la documentación establecida?</p> <p>_ No _ No sé _ Sí</p> <p>2- ¿Utilizaría usted la aplicación propuesta a la hora de generar la documentación referente al festival de artistas aficionados?</p> <p>_ No _ No sé _ Sí</p>

3- ¿Satisface sus necesidades, según el rol que desempeña en la gestión del movimiento de artistas aficionados, el sistema propuesto? _ 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 de gestión de la información estadística del festival? Argumente.
5- ¿Considera que se debería extender el uso del sistema propuesto al resto de la Universidad? Argumente.

.2.1. Resultados de las entrevistas

- Pregunta 1. De los 10 encuestados el 100 % consideran que No.
- Pregunta 2. De los 10 encuestados, el 100 % consideran que el sistema propuesto si satisfacen sus necesidades para la gestión del MAA.
- Pregunta 3. De los 10 encuestados, 8 de ellos consideran que les satisface mucho el uso del sistema propuesto y el resto considera que le da lo mismo.

Para las restantes 2 preguntas de la encuesta el total de la muestra coincidió que las funcionalidades que ofrece el sistema permite una correcta gestión del proceso.

.3. Historias de usuario

Tabla 10. Historia de usuario # 3

Historia de usuario	
Número: 3	Nombre: Listar unidades artísticas
Usuario: Todos los usuarios.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe listar las unidades artísticas existentes.	
Observaciones:	
<ul style="list-style-type: none"> • No es necesario estar autenticado. • Los datos a mostrar son nombre, manifestación, tiempo de duración e instructor. 	

Continúa en la próxima página

Tabla 10. Continuación de la página anterior

Interfaz:

Unidades Artistas				
Nombre	Manifestacion	Manifestacion	Duración	Instructor
UCI Dance dos	Música	Latinoamericana	11 minutos	Aránzazu Pol
UA del coreo	Música	Tradicional cubana	3 minutos	Aránzazu Pol

Tabla 11. Historia de usuario # 4

Historia de usuario	
Número: 4	Nombre: Modificar unidad artística
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
<p>Descripción: El sistema debe permitir modificar una unidad artística a través de un formulario donde el usuario rellena todos los campos.</p> <ul style="list-style-type: none"> • Nombre: Campo de texto. Admite caracteres especiales, letras y números. • Género: Campo de selección. • Manifestación: Campo de selección. • Provincia: Campo de selección. • Obra: Campo de texto. Admite caracteres especiales, letras y números. • Autor: Campo de texto. Admite caracteres especiales, letras y números. • Acompañamiento: Campo de texto. Admite caracteres especiales, letras y números. • Duración: Campo de texto. Solo admite números. • Recursos: Campo de texto. Admite caracteres especiales, letras y números. • Miembros: Campo de selección. Se podrá seleccionar más de uno. • Instructor: Campo de selección. Se podrá seleccionar solo uno. • Link: Campo de texto. Solo admite URL. 	

Continúa en la próxima página

Tabla 11. Continuación de la página anterior

Observaciones:

- El usuario debe estar autenticado con el rol correspondiente.
- Si el usuario introduce la información de forma incorrecta, el sistema emite un mensaje notificando el error.
- Si el usuario introduce la información dejando campos obligatorios vacíos, el sistema emite un mensaje indicándole que los campos obligatorios deben llenarse.
- si es Artes Visuales se permiten 10 miembros, si es Música o Danza se permiten 20 miembros donde hasta el 30 por ciento puede ser egresado. El sistema debe permitir ubicar link de encuesta de satisfacción de la unidad artística.

Interfaz:

Tabla 12. Historia de usuario # 5

Historia de usuario	
Número: 5	Nombre: Eliminar unidad artística
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe permitir eliminar una unidad artística a través de un botón.	

Continúa en la próxima página

Tabla 12. Continuación de la página anterior

Observaciones: Es necesario estar autenticado.
Interfaz:

Unidades Artistas

Nombre	Manifestacion	Manifestacion	Duración	Instructor	
UCI Dance dos	Música	Latinoamericana	11 minutos	Aránzazu Pol	Eliminar
UA del coreo	Música	Tradicional cubana	3 minutos	Aránzazu Pol	Eliminar

Tabla 13. Historia de usuario # 6

Historia de usuario	
Número: 6	Nombre: Generar PDF de la planilla de la unidad artística
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe permitir exportar PDF de inscripción de una unidad artística.	
Observaciones: Es necesario estar autenticado.	
Interfaz:	

ANEXO 6 PLANILLA OFICIAL DE INSCRIPCIÓN
COMITÉ ORGANIZADOR NACIONAL
PLANILLA OFICIAL DE INSCRIPCIÓN

Provincia : La Habana CES de Procedencia: UCI
Manifestación: Música Género: Popular
Nombre de la Unidad Artística:UCI MUSIC
Total Integrantes: 6 Femenino: Masculino:
Profesor - Instructor/Director: Jesusa Tudela Teléfono (s): -----
No. C.I.: 38303431828 Correo electrónico: javimanso@example.org
Obra seleccionada: : kjasfhkashnd Autor:
Tiempo de duración: 8 Se acompaña con: None
Necesidades Técnicas Básicas: 2 SILLAS

RELACION DE INTEGRANTES					
No	Nombres	Carrera	Año	No. de CI	Correo electrónico
40621165743	Aránzazu Pol	-	-	40621165743	benitezcustodia@example.com
98070165743	Alejandro Campmany Cejas	-	-	98070165743	alejandrocc@estudiantes.uci.cu

(Nombre completo y firma)

Vicedecano Facultad

(Nombre completo y firma)

Presidente de la FEU

Tabla 14. Historia de usuario # 7


Historia de usuario	
Número: 7	Nombre: Mostrar detalles de la unidad artística
Usuario: Todos los usuarios	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe mostrar los detalles una unidad artística.	
Observaciones: No necesario estar autenticado.	
Interfaz: <div style="text-align: center; margin-top: 20px;">  <p>UCI MUSIC</p> <p>Manifestacion: Música Instructor: Jesusa Tudela Miembros:</p> <ul style="list-style-type: none"> • Aránzazu Pol • Alejandro Campmany Cejas • Jesusa Tudela <p>Duracion: 8 minutos Provincia: La Habana Centro IES: UCI Genero: Popular Recursos: 2 SILLAS Obra: kjasfhkashnd</p> </div>	

Tabla 15. Historia de usuario # 8

Historia de usuario	
Número: 8	Nombre: Buscar unidades artísticas
Usuario: Todos los usuarios	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: :El sistema debe permitir buscar una unidad artística en la sección donde se listan las unidades artísticas.	
Observaciones: No será necesario autenticarse para acceder a dicha funcionalidad.	

Continúa en la próxima página

Tabla 15. Continuación de la página anterior

Interfaz:



Nombre	Manifestacion	Manifestacion	Duración	Instructor
UCI Dance dos	Música	Latinoamericana	11 minutos	Aránzazu Pol
UA del coreo	Música	Tradicional cubana	3 minutos	Aránzazu Pol

Tabla 16. Historia de usuario # 9

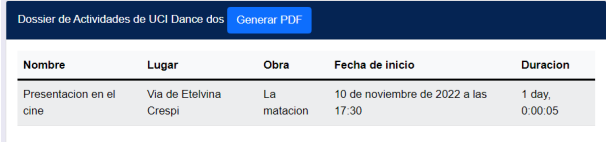
Historia de usuario	
Número: 9	Nombre: Mostrar dossier de actividades de la unidad artística
Usuario: Todos los usuarios	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe mostrar la relación de actividades de una unidad artística.	
Observaciones: <ul style="list-style-type: none"> • Es necesario estar autenticado. • Los campos son nombre, lugar , obra , unidad artística, fecha de inicio y duración. 	
Interfaz: 	

Tabla 17. Historia de usuario # 10

Historia de usuario	
Número: 10	Nombre: Generar PDF de la planilla de la unidad artística
Usuario: Todos los usuarios	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe permitir exportar PDF el dossier de actividades de una unidad artística.	

Continúa en la próxima página

Tabla 17. Continuación de la página anterior

<p>Observaciones:</p> <ul style="list-style-type: none"> • Es necesario estar autenticado. 												
<p>Interfaz:</p> <div style="text-align: center;"> <p>Dossier de Actividades de UCI Dance dos</p> <table border="1"> <thead> <tr> <th>Nombre</th> <th>Lugar</th> <th>UA</th> <th>Obra</th> <th>Fecha de inicio</th> <th>Duracion</th> </tr> </thead> <tbody> <tr> <td>Presentacion en el cine</td> <td>Via de Estelvína Crespi</td> <td>UCI Dance dos</td> <td>La Obra</td> <td>10 de noviembre de 2022 a las 17:30</td> <td>1 day, 0:00:05</td> </tr> </tbody> </table> </div>	Nombre	Lugar	UA	Obra	Fecha de inicio	Duracion	Presentacion en el cine	Via de Estelvína Crespi	UCI Dance dos	La Obra	10 de noviembre de 2022 a las 17:30	1 day, 0:00:05
Nombre	Lugar	UA	Obra	Fecha de inicio	Duracion							
Presentacion en el cine	Via de Estelvína Crespi	UCI Dance dos	La Obra	10 de noviembre de 2022 a las 17:30	1 day, 0:00:05							

Tabla 18. Historia de usuario # 11

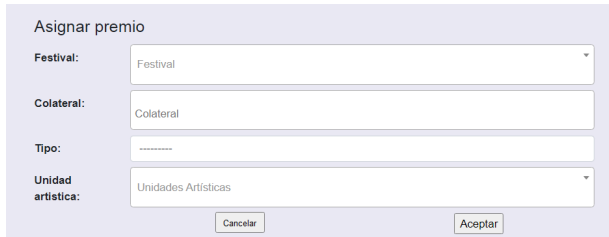
Historia de usuario	
Número: 11	Nombre: Asignar premio
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: :El sistema debe permitir asignar premio a una unidad artística a través de un formulario donde el usuario rellena todos los campos.	
<p>Observaciones:</p> <ul style="list-style-type: none"> • Es necesario estar autenticado. • Los campos son festival, unidad artística, tipo y premio colateral. 	
<p>Interfaz:</p> <div style="text-align: center;">  </div>	

Tabla 19. Historia de usuario # 12

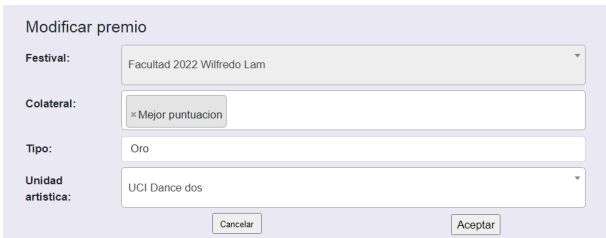
Historia de usuario	
Número: 12	Nombre: Modificar premio
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe permitir modificar un premio a través de un formulario donde el usuario rellena todos los campos.	
Observaciones: <ul style="list-style-type: none"> • Es necesario estar autenticado. 	
Interfaz:	
	

Tabla 20. Historia de usuario # 13


Historia de usuario	
Número: 13	Nombre: Eliminar premio
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe permitir eliminar un premio a través de un botón.	
Observaciones: <ul style="list-style-type: none"> • Es necesario estar autenticado. 	
Interfaz:	
	

Tabla 21. Historia de usuario # 14


Historia de usuario	
Número: 14	Nombre: Listar premios
Usuario: Todos los usuarios	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema deberá listar todos los premios empezando por los mas recientes.	
Observaciones: <ul style="list-style-type: none"> • No será necesario autenticarse para acceder a dicha funcionalidad. • En caso de estar autenticado se mostrará un botón para filtrar por los premios en los que ha participado dicho usuario. 	
Interfaz: <div style="text-align: center;">  </div>	

Tabla 22. Historia de usuario # 15


Historia de usuario	
Número: 15	Nombre: Buscar premio
Usuario: Todos los usuarios	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: :El sistema debe permitir buscar un premio en la sección donde se listan los premios.	
Observaciones: No será necesario autenticarse para acceder a dicha funcionalidad.	
Interfaz: <div style="text-align: center;">  </div>	

Tabla 23. Historia de usuario # 16

Historia de usuario	
Número: 16	Nombre: Notificar premio por correo a los miembros de la unidad artística
Usuario: Todos los usuarios	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe notificar de manera automática por correo a los miembros de una unidad artística cuando les sea asignado un premio.	
Observaciones: Se realiza de forma automática al asignar premio.	
Interfaz:	
<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;"> <p>La Dirección de Extensión Universitaria la informa que ha obtenido oro en el Festival 1</p> </div>	

Tabla 24. Historia de usuario # 17

Historia de usuario	
Número: 17	Nombre: Crear festival
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: :El sistema debe permitir crear un festival a través de un formulario donde el usuario rellena todos los campos.	
Observaciones:	
<ul style="list-style-type: none"> • Es necesario estar autenticado. 	

Continúa en la próxima página

Tabla 24. Continuación de la página anterior

Interfaz:

Crear festival

Nivel: Universidad

Año: 2022

Lugar: Wilfredo Lam

Jurado: xcvbkfghjkgf

Unidades Artísticas: UCI Dance dos, UA del coreo, Test Validator, UCI Dance

Cancelar Aceptar

Tabla 25. Historia de usuario # 18

Historia de usuario	
Número: 18	Nombre: Modificar festival
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe permitir modificar un festival a través de un formulario donde el usuario rellena todos los campos.	
Observaciones:	
<ul style="list-style-type: none"> • Es necesario estar autenticado. 	

Continúa en la próxima página

Tabla 25. Continuación de la página anterior

Interfaz:

Modificar festival

Nivel: Universidad

Año: 2022

Lugar: Wilfredo Lam

Jurado: xcvbkfghjkgf

Unidades Artísticas: UCI Dance dos, UA del coreo, Test Validator, UCI Dance

Cancelar Aceptar

Tabla 26. Historia de usuario # 19

Historia de usuario									
Número: 19	Nombre: Eliminar festival								
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.									
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja								
Puntos estimados: 1	Iteración asignada: 1								
Programador responsable: Alejandro Campmany Cejas									
Descripción: : El sistema debe permitir eliminar un festival a través de un botón.									
Observaciones:									
<ul style="list-style-type: none"> Es necesario estar autenticado. 									
Interfaz:									
<p>Festivales de artistas aficionados</p> <table border="1"> <thead> <tr> <th>Nivel</th> <th>Fecha</th> <th>Lugar</th> <th></th> </tr> </thead> <tbody> <tr> <td>Facultad</td> <td>2022</td> <td>Wilfredo Lam</td> <td>Eliminar</td> </tr> </tbody> </table>		Nivel	Fecha	Lugar		Facultad	2022	Wilfredo Lam	Eliminar
Nivel	Fecha	Lugar							
Facultad	2022	Wilfredo Lam	Eliminar						

Tabla 27. Historia de usuario # 20

Historia de usuario	
Número: 20	Nombre: Mostrar detalles del festival
Usuario: Todos los usuarios	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja

Continúa en la próxima página

Tabla 27. Continuación de la página anterior

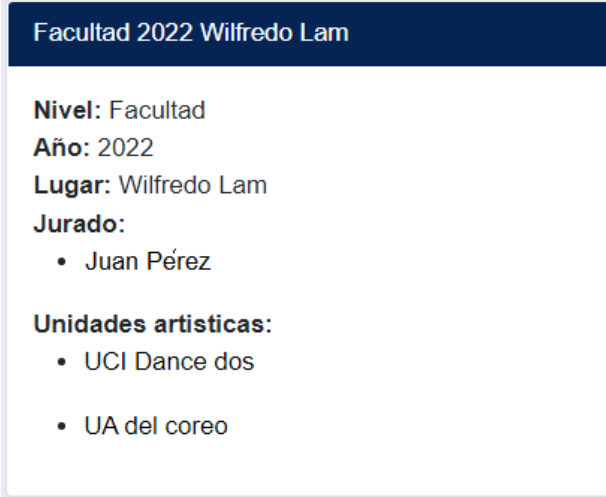
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: : El sistema debe mostrar los detalles del festival	
Observaciones:	
<ul style="list-style-type: none"> • No será necesario autenticarse para acceder a dicha funcionalidad. 	
Interfaz:	
 <p>The screenshot shows a dark blue header with the text 'Facultad 2022 Wilfredo Lam'. Below it, the following details are listed:</p> <ul style="list-style-type: none"> Nivel: Facultad Año: 2022 Lugar: Wilfredo Lam Jurado: <ul style="list-style-type: none"> • Juan Pérez Unidades artísticas: <ul style="list-style-type: none"> • UCI Dance dos • UA del coreo 	

Tabla 28. Historia de usuario # 21


Historia de usuario							
Número: 21	Nombre: Buscar festival						
Usuario: Todos los usuarios							
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja						
Puntos estimados: 1	Iteración asignada: 1						
Programador responsable: Alejandro Campmany Cejas							
Descripción: :El sistema debe permitir buscar un festival en la sección donde se listan los festivales.							
Observaciones: No será necesario autenticarse para acceder a dicha funcionalidad.							
Interfaz:							
 <p>The screenshot shows a dark blue header with the text 'Festivales de artistas aficionados' and a search bar with the placeholder text 'Buscar...'. Below the header, there is a table with the following data:</p> <table border="1"> <thead> <tr> <th>Nivel</th> <th>Fecha</th> <th>Lugar</th> </tr> </thead> <tbody> <tr> <td>Facultad</td> <td>2022</td> <td>Wilfredo Lam</td> </tr> </tbody> </table>		Nivel	Fecha	Lugar	Facultad	2022	Wilfredo Lam
Nivel	Fecha	Lugar					
Facultad	2022	Wilfredo Lam					

Tabla 29. Historia de usuario # 22


Historia de usuario	
Número: 22	Nombre: Insertar evidencias (foto, video o documento) en el festival.l
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe permitir insertar las evidencias asociadas a cada festival a través de un formulario.	
Observaciones: <ul style="list-style-type: none"> • Es necesario estar autenticado. 	
Interfaz: 	

Tabla 30. Historia de usuario # 23

Historia de usuario	
Número: 23	Nombre: Eliminar evidencias (foto, video o documento) en el festival.l
Usuario: Encargado de facultad, Encargado de universidad, Encargado de provincia y Encargado nacional.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe permitir eliminar las evidencias asociadas a cada festival a través de un botón.	
Observaciones: <ul style="list-style-type: none"> • Es necesario estar autenticado. 	

Continúa en la próxima página

Tabla 30. Continuación de la página anterior



Tabla 31. Historia de usuario # 24

Historia de usuario	
Número: 24	Nombre: Mostrar evidencias (foto, video o documento) en el festival.
Usuario: Todos los usuarios.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe permitir eliminar las evidencias asociadas a cada festival a través de un formulario.	
Observaciones: <ul style="list-style-type: none"> • No es necesario estar autenticado para acceder a dicha funcionalidad. • Se mostraran las evidencias para cada festival con una descripción debajo. 	
Interfaz:	

Tabla 32. Historia de usuario # 25

Historia de usuario	
Número: 25	Nombre: Mostrar programa artístico por manifestación
Usuario: Todos los usuarios.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe mostrar los programas artísticos de cada manifestación.	

Continúa en la próxima página

Tabla 32. Continuación de la página anterior


<p>Observaciones:</p> <ul style="list-style-type: none"> • No es necesario estar autenticado para acceder a dicha funcionalidad. • Los campos son nombre, lugar, unidad artística, jurado , fecha de inicio y duración.
<p>Interfaz:</p> 

Tabla 33. Historia de usuario # 26


Historia de usuario	
Número: 26	Nombre: Generar PDF del programa artístico por manifestación
Usuario: Todos los usuarios.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe permitir exportar en formato PDF los programas artísticos de cada manifestación	
Observaciones: No es necesario estar autenticado para acceder a dicha funcionalidad.	
<p>Interfaz:</p> 	

Tabla 34. Historia de usuario # 27

Historia de usuario	
Número: 27	Nombre: Autenticar usuario
Usuario: Todos los usuarios.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Alejandro Campmany Cejas	
Descripción: El sistema debe permitir autenticar usuario a través de un formulario donde el usuario rellena todos los campos.	

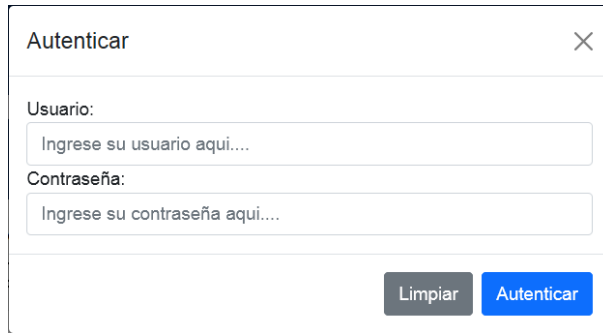
Continúa en la próxima página

Tabla 34. Continuación de la página anterior

Observaciones:

- No es necesario estar autenticado para acceder a dicha funcionalidad.
- Los campos son usuario y contraseña.

Interfaz:



The image shows a screenshot of a web application's authentication dialog box. The dialog has a title bar with the text 'Autenticar' and a close button (X) in the top right corner. Below the title bar, there are two input fields. The first is labeled 'Usuario:' and contains the placeholder text 'Ingrese su usuario aqui...'. The second is labeled 'Contraseña:' and contains the placeholder text 'Ingrese su contraseña aqui...'. At the bottom right of the dialog, there are two buttons: a grey button labeled 'Limpiar' and a blue button labeled 'Autenticar'.