



UNIVERSIDAD DE CIENFUEGOS
INGENIERÍA
DEPARTAMENTO DE INFORMÁTICA

Universidad de Cienfuegos

“Carlos Rafael Rodríguez”

Facultad de Ingeniería

Título:

Plataforma Audiovisual y Aprendizaje Académico

Autores:

Sheila Gabriela Pérez Soto

David Padrón García

Tutor(es):

Ing. Laura Romero Romero

MSc. Richard Darian Sánchez Rivero

Cienfuegos, Cuba

Curso: 2023

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad Cienfuegos los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

(Autor)

(Autor)

(Tutor)

(Cotutor)

Índice

Introducción.....	1
Capítulo 1: FUNDAMENTACIÓN TEÓRICA	9
Introducción	9
1.1 Referentes teóricos asociados al dominio del problema	9
1.2 Uso de las TIC en el proceso de enseñanza y aprendizaje	10
1.3 Soporte para sistemas informáticos	11
1.4 Metodología de desarrollo de software	13
1.5 Arquitectura de software	15
1.5.1 Estilos arquitectónicos	16
1.6 Tecnologías y herramientas para el desarrollo de la solución	17
1.7 Conclusiones parciales	19
Capítulo 2: Propuesta de solución	20
2.1 Concepción general del sistema	20
2.1.1 Requisitos Funcionales.....	21
2.1.2 Requisitos no funcionales.....	26
2.2 Historias de Usuario.....	27
2.3 Descripción de la arquitectura.....	28
2.3.1 Patrón de arquitectura Modelo Vista Plantilla (MTV)	29
2.4 Diagrama de clases de diseño	30
2.5 Patrones de diseño utilizados	31
2.6 Modelo de Datos	33
2.7 Estándar de codificación empleado	33
2.8 Conclusiones parciales	34
Capítulo 3: VALIDACIÓN DE LA VIABILIDAD DE LA PROPUESTA DE SOLUCIÓN	35

3.1 Pruebas de software	35
3.1.1 Pruebas de caja blanca	35
3.1.2 Pruebas de caja negra.....	38
3.2 Diseño de pruebas funcionales.....	42
3.3 Conclusiones parciales	45
Conclusiones generales	47
Recomendaciones.....	48
Referencias	49

Índice de tablas

Tabla 1.Requisitos Funcionales	26
Tabla 2.Historias de Usuario	28
Tabla 3.Estrategia de prueba aplicada a la solución.	35
Tabla 4.Prueba funcional para el caso de uso Agregar Películas.	43
Tabla 5. Prueba funcional para el caso de uso Agregar Serie.	45

Índice de figuras

Figura 1. Arquitectura Modelo Vista Template	29
Figura 2.Aplicación de la arquitectura a la solución propuesta	30
Figura 3. Diagrama de clases del diseño	31
Figura 4.Modelo físico de datos	33
Figura 5.Funcionalidad login_view(loguear usuario).	36
Figura 6 Grafo resultante de la función login_view	37
Figura 7 Caso de prueba 1.....	40
Figura 8 Caso de prueba 2.....	40
Figura 9 Caso de prueba 3.....	41
Figura 10 Caso de prueba 4.....	41
Figura 11 Caso de prueba 5.....	41
Figura 12 Caso de prueba 6.....	41
Figura 13 Caso de prueba 7.....	42
Figura 14 Caso de prueba 9.....	42

Resumen

El presente trabajo se desarrolló en la Universidad de Cienfuegos. La creación de una plataforma audiovisual y de aprendizaje académico surge como una necesidad crucial para complementar las herramientas existentes en la institución, especialmente ante la creciente demanda de contenido multimedia en el sector educativo. A pesar de contar con Moodle como herramienta de aprendizaje, se identificó la falta de una plataforma específica que integre contenidos audiovisuales y recursos educativos interactivos, capaces de enriquecer y potenciar la experiencia educativa de los estudiantes universitarios. La plataforma fue diseñada pensando en la accesibilidad dentro de la universidad, pudiendo ser desplegada y experimentada en una amplia gama de dispositivos, desde ordenadores de escritorio hasta dispositivos móviles, lo que garantiza su uso generalizado en la Universidad de Cienfuegos y su accesibilidad para un mayor número de usuarios, en consonancia con las tendencias actuales de educación en línea. En el proceso de desarrollo del sistema se emplearon diversas tecnologías y herramientas, incluyendo Django, Python, HTML5, CSS, Javascript, así como sistemas de gestión de bases de datos como PostgreSQL y PgAdmin, entre otras, con el fin de crear una plataforma moderna.

Palabras claves: plataforma, audiovisual, aprendizaje, enriquecer, potenciar.

Abstract

This work was developed at the University of Cienfuegos. The creation of an audiovisual and academic learning platform emerges as a crucial need to complement the existing tools in the institution, especially given the growing demand for multimedia content in the educational sector. Despite having Moodle as a learning tool, the lack of a specific platform that integrates audiovisual content and interactive educational resources, capable of enriching and enhancing the educational experience of university students, was identified. The platform was designed with accessibility within the university in mind, and can be deployed and experienced on a wide range of devices, from desktop computers to mobile devices, which guarantees its widespread use at the University of Cienfuegos and its accessibility to a wider audience. number of users, in line with current trends in online education. Various technologies and tools were used in the system development process, including Django, Python, HTML5, CSS, Javascript, as well as database management systems such as PostgreSQL and PgAdmin, among others, in order to create a modern platform.

Key words: platform, audiovisual, learning, enriching, empowering.

Introducción

Las tecnologías de la información y comunicación (TIC) son el resultado de poner en interacción la informática y las telecomunicaciones. Todo, con el fin de mejorar el procesamiento, almacenamiento y transmisión de la información. El término TIC también se usa para referirse a la convergencia de redes audiovisuales y telefónicas con redes informáticas a través de un único sistema de cableado u enlace. Existen grandes incentivos económicos para fusionar la red telefónica con el sistema de red informática utilizando un único sistema unificado de cableado, distribución de señales y gestión. TIC es un término general que incluye cualquier dispositivo de comunicación, que abarca radio, televisión, teléfonos celulares, computadoras y hardware de red, sistemas satelitales, etc., así como los diversos servicios y dispositivos con ellos, tales como videoconferencias y aprendizaje a distancia[1].

Hoy en día no se puede hablar de eficiencia o novedad si no existe una aplicación y correcta utilización de la tecnología moderna, y es que se puede asegurar que las TIC tienen la respuesta efectiva a la mano. Las TIC han afectado todas las áreas del desarrollo social y se han constituido en elementos sustantivos inherentes al desarrollo de todas las esferas de la vida. Su introducción en la sociedad va acompañada de grandes beneficios, pero no está exenta de problemas.

Actualmente podemos notar que las TIC son una de las herramientas más utilizadas a nivel mundial, puesto que los medios informáticos que se utilizan en las aulas de clase benefician a los estudiantes en diversos factores. Estas han traído consigo un sinnúmero de recursos digitales que permiten la facilidad en la gestión educativa y empresarial.

El intercambio de información importante en internet, la resolución de los problemas que se presentan día a día y la discusión de temas relevantes sobre situaciones que suceden en el mundo son algunos de los aprendizajes que podemos obtener con la utilización de la tecnología y la comunicación.

En este contexto, donde es vital el aprovechamiento de las TIC como herramientas para el desarrollo del conocimiento, la economía y la actividad político-ideológica.

El uso de las Tecnologías de la Información y las Comunicaciones (TIC) y de Internet 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 nuestros 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 tecnologías de la Informática y las Comunicaciones para satisfacer las necesidades de información y conocimiento de la sociedad.[2]

Es por ello que en nuestro país el centro de la estrategia de informatización es el hombre y que los objetivos prioritarios de esta estrategia son los sectores de la Salud Pública y la Educación, para potenciar los logros que en estas áreas ha logrado nuestra Revolución. Desde la década de 1970 se identifica en Cuba la necesidad de la participación activa y coherente de las Instituciones de la Educación Superior (IES) en el desarrollo de las tecnologías, lo que se materializa con la creación de carreras afines a la Computación y la Electrónica. Parte importante del éxito en este proceso está relacionado con la creación de la Unión de Informáticos de Cuba, una organización que nacerá para sumar todas las miradas posibles de especialistas del sector a lo largo y ancho de la geografía nacional. La creación en 1987 de los Jóvenes Club de Computación y Electrónica; de INFOMED en 1992; de la Universidad de las Ciencias Informáticas (UCI) en el año 2002 con el objetivo fundamental de la formación de recursos humanos en este campo; del Ministerio para la Informática y las Comunicaciones, que ha constituido una estrategia vital para el desarrollo de las TIC en Cuba. Un hito importante en este proceso fue la creación en 1979 de la Universidad de Cienfuegos, la cual tiene la misión de gestionar mediante el conocimiento, la innovación y la participación inclusiva, la formación integral y continua de profesionales competentes y comprometidos con la Revolución Socialista, para la dinamización del desarrollo socioeconómico sostenible del territorio [3].

El Ministerio de Educación Superior también ha montado una estrategia la cual incluye cambios certeros en la tecnología informática realizando una inversión todos los años que ha permitido que las universidades cuenten con laboratorios de alta tecnología, no obstante, por el creciente número de alumnos que ingresan a nuestras universidades aún es insuficiente, realizándose inversiones todos los años en este campo.

Las universidades cubanas están conectadas a Internet. Lo mismo ocurre con los centros científicos, órganos de prensa y otras instituciones que han sido priorizadas en la creación de las condiciones técnicas a que se ha podido llegar, no obstante, los obstáculos propios de una nación subdesarrollada, bloqueada y con pocos recursos financieros.

La investigación académica se ha visto ayudada enormemente por INTERNET, principalmente en las Universidades y la creación o la utilización de modelos informáticos pueden ser una herramienta novedosa y muy atractiva para despertar el interés cognitivo de los estudiantes.

Desde diciembre de 1999 comenzó la utilización masiva de medios audiovisuales en las escuelas cubanas, lo cual generó la necesidad de producir programas de televisión destinados a los centros escolares. Para asegurar la recepción de estos programas, todos los centros de enseñanza general están dotados con un televisor en cada aula como asesores del programa audiovisual. El 100% de los centros educacionales del país usan las TIC como apoyo a los programas de clases y utilizan las aplicaciones de los Software Educativos.

El impacto de las Tecnologías de la Información y Comunicación (TIC) sobre la educación, propicia posiblemente uno de los mayores cambios en el ámbito de la Educación. A través de Internet y de las informaciones y recursos que ofrece, en el aula se abre una nueva ventana que nos permite acceder a múltiples recursos, informaciones y comunicarnos con otros, lo que nos ofrece la posibilidad de acceder con facilidad a conocer personalidades de opiniones diversas. Por otro lado, las nuevas teorías de aprendizaje que centran su atención no tanto en el profesor y el proceso de enseñanza,

como en el alumno y el proceso de aprendizaje, tienen un buen aliado en estos medios, si se utilizan atendiendo a los postulados del aprendizaje socio constructivo y bajo los principios del aprendizaje significativo.

Como han demostrado diferentes estudios, el uso de las TIC en la educación depende de múltiples factores (infraestructuras, formación, actitudes, apoyo del equipo directivo, etc.), entre los cuales el más relevante es el interés y la formación por parte del profesorado, tanto a nivel instrumental como pedagógico. La red de informática del ministerio de Educación nació con el concepto de llevar la informatización y el trabajo en red de los centros educativos del país, usando los laboratorios docentes como punto común para la interacción tanto de los estudiantes como los profesores.

La introducción de las TIC en el proceso de enseñanza-aprendizaje tiene gran repercusión en la expansión de procesos formativos que utilizan la modalidad a distancia y semipresencial pues la posibilidad de aumento de la interacción entre el profesor o tutor y el estudiante, la posibilidad de acceso en el momento y lugar deseado, la adecuación a las características personales de los que se benefician del programa de formación. Otras características, como el aumento de información y la posibilidad de acceso, de comunicación del estudiante con varios agentes y contextos culturales, no solo con el profesor o tutor, son favorecedores de este proceso. Por otra parte, las TIC con su desarrollo creciente propicia la combinación y utilización de los diferentes sentidos lo que favorece la motivación, la memorización y de forma general el aprendizaje. Algunos autores plantean respecto al impacto de las TIC en la educación los siguientes elementos:

- Surgimiento de nuevas competencias tecnológicas.
- La posibilidad de encontrar vías de aprendizaje fuera de las instituciones formales.
- Usos de las TIC en educación.
- Necesidad de formación continua.
- Nuevos entornos de aprendizaje y de modelos pedagógicos.

- Los impactos identificados se pueden observar en las funciones fundamentales de las TIC en la educación:
- Fuente de información (hipermedial).
- Canal de comunicación interpersonal y para el trabajo colaborativo y para el intercambio de información e ideas (correos electrónicos, foros telemáticos).
- Medio de expresión y para la creación (procesadores de textos y gráficos, editores de páginas web y presentaciones multimedia, cámara de vídeo).
- Instrumento cognitivo y para procesar la información: hojas de cálculo, gestores de bases de datos.
- Instrumento para la gestión, ya que automatizan diversos trabajos de la gestión de los centros: secretaría, acción tutorial, asistencias, bibliotecas.
- Recurso interactivo para el aprendizaje. Los materiales didácticos multimedia informan, entrenan, simulan, guían aprendizajes, motivan.
- Medio lúdico y para el desarrollo psicomotor y cognitivo.[4]

En nuestro centro existe falta de acceso a recursos educativos y de entretenimiento para los estudiantes de la universidad afectando negativamente su capacidad para aprender y mantenerse motivados en sus estudios, así como su calidad de vida en general. La plataforma existente "Moodle" presenta varias desventajas como por ejemplo los archivos que se suben tienen que ser de poca capacidad, lo cual no permite que los profesores tengan amplitud en cuanto a la motivación de los estudiantes para presentarles videos educativos o películas de interés que los mismos puedan descargar e interiorizar. Tampoco posee ningún segmento recreativo que motive al estudiante a acceder más a menudo a este sitio. Sería importante encontrar una solución alternativa para proporcionar a los estudiantes acceso a estos recursos, ya sea a través de otras plataformas en línea o mediante la creación de una aplicación web propia.

Como se evidencia en lo antes descrito, la disponibilidad de una plataforma virtual de audiovisuales, no existe. En la actualidad, este proceso presenta un conjunto de carencias en la universidad, que inciden negativamente en la enseñanza-aprendizaje y entretenimiento.

A partir de la problemática antes descrita se identifica como **problema a resolver**: será el desarrollo de una aplicación web que permita a los estudiantes y profesores una mejor comodidad a la hora del estudio independiente y la preparación en cuanto a cultura general a través de los canales informativos, deportivos y recreativos.

Este problema se enmarca en el **objeto de Estudio**: el diseño y desarrollo de una aplicación web que satisfaga las necesidades de los estudiantes y profesores de la universidad de Cienfuegos en cuanto a películas y series, y aprendizaje estudiantil.

Su **campo de acción**: proceso de diseño, desarrollo, implementación y mantenimiento de la aplicación web para satisfacer las necesidades de los usuarios de la universidad de Cienfuegos.

Para darle solución al problema, se plantea como **objetivo general**: desarrollar una aplicación web que sea fácil de usar, segura y que ofrezca una amplia variedad de contenido de películas y series, así como herramientas de aprendizaje estudiantil que sean útiles para los usuarios, desglosado en los siguientes **objetivos específicos**:

- Identificar las necesidades y preferencias de los usuarios de la aplicación.
- Diseñar una interfaz de usuario intuitiva y atractiva.
- Desarrollar una aplicación web que ofrezca una amplia variedad de contenido de películas y series.
- Realizar pruebas y depuración de la aplicación para asegurar su correcto funcionamiento.

Idea a defender: la falta de una aplicación web de películas y series y de aprendizaje estudiantil en la universidad de Cienfuegos limita el acceso de los estudiantes y profesores a recursos educativos y de entretenimiento, lo que puede afectar negativamente su rendimiento académico y su calidad de vida. Por lo tanto, la creación de una aplicación web que facilite estas necesidades puede mejorar la experiencia de los usuarios y contribuirá al éxito académico y personal de la comunidad universitaria.

El **método Científico de Investigación** es la forma de estudiar los fenómenos de la naturaleza y la sociedad para descubrir sus relaciones y su esencia, el mismo se puede clasificar en métodos de nivel teórico o empírico los cuales están relacionados entre sí de forma dialéctica. A continuación, se muestran los métodos utilizados para el desarrollo de esta investigación.

Métodos del nivel teórico

- Método Analítico-Sintético: Posibilitó realizar un análisis de las distintas partes que afectan el objeto de estudio y sintetizar los elementos más significativos.
- Método Histórico-Lógico: Para la realización de la investigación se hizo necesario estudiar la evolución del problema y la existencia de metodologías, procedimientos y sistemas informáticos similares al que se pretende elaborar; determinando cuáles son las tendencias actuales para el desarrollo de sistemas de gestión académica.

Métodos del nivel empírico

- Observación: Se empleó para identificar características en el proceso de disponibilidad de audiovisuales, quiénes intervienen, qué utilizan. De igual manera se empleó para identificar las carencias más significativas en este proceso.

El documento de presentación de los resultados de la investigación está estructurado en 3 capítulos:

Capítulo 1 Fundamentos teórico-metodológicos: Se presentan los elementos teóricos referente al desarrollo de plataformas audiovisuales, que sirven de base a la investigación del problema planteado. En este capítulo se analizarán los principales conceptos relacionados con el objeto de estudio, se realiza una descripción de los sistemas existentes de desarrollo web; y una explicación de herramientas y tecnologías potenciales a utilizar en el desarrollo de la propuesta de solución.

Capítulo 2 Descripción de la propuesta de solución: Se describe la propuesta de solución y los artefactos generados por el escenario número 4 en la metodología Agile Unified Process (AUP).

Capítulo 3 Validación de la viabilidad de la propuesta de solución: Se valora la viabilidad de la propuesta de solución mediante la validación de los requisitos y el diseño propuesto, además de la realización de pruebas funcionales, unitarias y de aceptación de la plataforma de audiovisuales “Visuales UCF”.

Capítulo 1: FUNDAMENTACIÓN TEÓRICA

Introducción

El presente capítulo muestra una visión general del trabajo propuesto y aborda los aspectos básicos del tema en cuestión. Se analizarán las características y beneficios de su utilización, los cuales contribuyen al soporte teórico del proyecto; fundamenta el uso del lenguaje de programación y las herramientas y utilizadas; y argumenta sobre la metodología de desarrollo seleccionada.

1.1 Referentes teóricos asociados al dominio del problema

Plataforma

En la actualidad, existen muchas plataformas digitales con enfoques especializados en contextos profesionales, de negocios, de entretenimiento, educativos, de salud, entre otros, las cuales han venido tomando cada vez más popularidad en el ámbito empresarial y de los negocios, volviéndose así una excelente oportunidad para materializar ideas de negocio[5].

Audiovisual

Audiovisual, en su concepto, es la integración e interrelación plena entre lo auditivo y lo visual para producir una nueva realidad o lenguaje. La percepción es simultánea. Se crean así nuevas realidades sensoriales mediante mecanismos como la armonía (a cada sonido le corresponde una imagen), complementaria (lo que si aporta lo visual lo aporta lo auditivo), refuerzo (se refuerzan los significados entre sí) y contraste (el significado nace del contraste entre ambos)[6].

Medios de enseñanza

Los medios de enseñanza y aprendizaje. Responden al ¿con qué enseñar y con qué aprender? y pueden considerarse objetos naturales, conservados o sus representaciones, materiales, instrumentos o equipos que forman parte de la actividad de docentes y estudiantes.

Permiten la facilitación del proceso, a través de objetos reales, sus representaciones e instrumentos sirven de apoyo material para la apropiación del contenido,

complementando el método, para la consecución de los objetivos propuestos por el docente. Dentro de las funciones más importantes de los mismos tenemos:

- Actúan en el proceso de comunicación, en el cual están representado por el canal a través del cual se envía el mensaje.
- Favorecen la formación de convicciones, hábitos, habilidades y normas de conductas en los estudiantes.
- Motivan el aprendizaje y aumentan la concentración de la actuación.
- Aumentan la efectividad del proceso docente al mejorar la calidad de la enseñanza, sistematizándola y empleando menos tiempo y esfuerzo.
- Permiten el control del proceso pedagógico.

Los medios audiovisuales deben contribuir significativamente a la permanente educación del pueblo; deben destinarse fundamentalmente a incrementar el nivel educativo, cultural, político y orientar y promover actitudes que ayuden al desarrollo económico y social de toda la población. No es un secreto para nadie que estos medios de enseñanza permiten activar los mecanismos que facilitarán un mejor proceso del conocimiento ya que no solo enriquece la sensopercepción de los objetos, fenómenos y procesos de estudio, sino que también estimulan la motivación y el interés por aprender, a la vez que ahorran tiempo y esfuerzo durante el proceso educativos[7].

1.2 Uso de las TIC en el proceso de enseñanza y aprendizaje

El uso de las Tecnologías de la Información y la Comunicación (TIC) en el proceso de enseñanza-aprendizaje tiene como propósito¹:

- Que el profesor promueva el trabajo colaborativo utilizando TIC.
- Proveer de herramientas para que el profesor utilice diversas estrategias.
- Ampliar los ambientes de enseñanza-aprendizaje.
- Que los alumnos construyan el conocimiento.

La incorporación de las TIC en el proceso de enseñanza-aprendizaje busca incentivar al estudiante para lograr su propio aprendizaje y desarrollar habilidades y destrezas para ejecutar sus ideas. Las TIC tienen funciones como ser un medio de expresión, fuente abierta de información, instrumentos para procesar la información y construir nuevos conocimientos de aprendizajes, canal de comunicación presencial, canal de distribución

virtual, instrumentos para la gestión medio didáctica³. Las TIC han propiciado grandes cambios en la sociedad y en la educación, que va desde conocimientos nuevos hasta cambio de nuevas técnicas y estrategias y han facilitado modificar tanto los ambientes de aprendizajes como métodos para enseñar y aprender. Las TIC permiten la participación y el empoderamiento de los educandos, generando nuevas experiencias formativas en las cuales se crean escenarios más flexibles e inclusivos, que les ayuden a crear y fortalecer sus habilidades de forma participativa[8].

1.3 Soporte para sistemas informáticos

Internacionales

Platzi es una de las plataformas de educación profesional en línea más grandes de América Latina, fundada en 2011 por Freddy Vega y Christian Van der Henst. Hoy posee 3 millones de estudiantes activos en más de 140 países, miles de cursos y áreas de estudio enfocadas en el desarrollo de habilidades y conocimientos que requieren las grandes empresas del mundo.

Propone una forma novedosa de aprendizaje que busca romper el ciclo de pobreza y crear un impacto positivo en la economía de la región.

Su metodología de enseñanza está enfocada en crear y fortalecer habilidades de mayor demanda laboral. Algunas de sus áreas de estudio son:

- Ingeniería y desarrollo
- Diseño UX/UI
- Inglés
- Marketing digital
- Ciencias de datos
- Emprendimiento y negocios
- Habilidades blandas
- Periodismo digital
- Producción audiovisual[9]

Udemy es una plataforma de aprendizaje en línea. Está dirigido a adultos profesionales. A diferencia de los programas académicos MOOC conducidos por tradicionales cursos

de trabajo creados por las universidades, Udemty utiliza contenido de creadores en línea para vender y así conseguir ganancias. Proporciona herramientas para que los usuarios puedan crear un curso, promocionarlo y ganar dinero con gastos de inscripción de los estudiantes.

Ninguno de los cursos de Udemty es actualmente equivalentes a la obtención de un título universitario. Los estudiantes toman cursos en gran parte simplemente como medio para mejorar sus habilidades. Algunos cursos generan crédito hacia certificación técnica. Udemty ha hecho un esfuerzo especial para atraer a empresas que buscan crear cursos de trabajo especialmente para empleados de su compañía.[10]

Netflix es un servicio de streaming por suscripción que les permite a sus miembros ver series y películas en un dispositivo con conexión a internet.[11]

Plex es un servicio totalmente gratuito para ver películas y series que te permite utilizar un ordenador como servidor para acceder al contenido que añadas a tu biblioteca desde cualquier sitio, desde otros ordenadores hasta directamente tu televisor. Y tiene un modo de pago con opciones extra.[12]

Nacionales

Picta es una servicio de streaming de video bajo demanda como televisión en vivo, de Cuba, y que también está disponible en Latinoamérica. Es desarrollada por la Universidad de las Ciencias Informáticas, fue lanzado fines del 2018 como el primer servicio VOD originario de la isla.

Es la plataforma líder de video bajo demanda de Cuba, para registrarte es mediante líneas móviles de Cuba (únicamente por el momento solo en la isla), a partir de marzo del 2020 se agrega canales en vivo tales como Cubavisión, Cubavisión Internacional, Telerebelde y Canal Caribe.[13]

1.4 Metodología de desarrollo de software

Las metodologías de desarrollo de software, se definen como un conjunto de consideraciones y parámetros, cuyo objetivo es gestionar el proceso de desarrollo de un sistema informático. Dichas consideraciones abarcan desde herramientas, modelos, he incluso otras metodologías, necesarias dentro del marco de trabajo del proyecto.

Cada metodología de desarrollo de software está diseñada para un tipo de proyecto en específico. Para ello se toman en cuenta diversos factores. Desde aspectos técnicos como las herramientas y los dispositivos con los que se trabajarán. Hasta la organización de las tareas y el equipo de trabajo con el que se cuenta.

A lo largo de los años, han surgido diferentes tipos de metodología de desarrollo de software. Cada una, proponiendo un esquema de trabajo diferente. De entre estas sobresalen tanto las metodologías tradicionales como las ágiles. Las metodologías tradicionales, también conocidas como modelos de proceso prescriptivo. Se crearon con el propósito de mejorar el pésimo tratamiento que se le daba al software durante sus primeros años de vida. Estas se caracterizan por su poca flexibilidad a los cambios. Lo que les ha dado la mala fama.

Las metodologías tradicionales se rigen bajo una estructura secuencial inalterable basada en etapas. Dichas etapas son: el análisis de requerimientos, diseño, programación, pruebas y mantenimiento. Se les da gran importancia a los requerimientos del proyecto. Dado que se precisa de un amplio estudio para su definición, y una vez establecidos, no pueden alterarse de ninguna manera.

Por otra parte, las metodologías ágiles se caracterizan por su versatilidad debido a lo sencillo que les resulta el adaptarse a los cambios. Sin alterar las características originales del proyecto en el proceso. Lográndose un control mucho más dinámico del mismo, amén de una mejor respuesta a las necesidades tanto de los clientes, como del mercado general.

En principio, estas surgen como respuesta a las carencias de las metodologías tradicionales. Sin embargo, gracias a los constantes cambios en los mercados y los

entornos de trabajo. Estas han ido evolucionando, en favor de un proceso de desarrollo optimizado. Mejorando la producción, a la vez que se reducen costos y tiempo.

Teniendo en cuenta las características del equipo de desarrollo (2 miembros + cliente), la participación activa de los clientes en la solución y el carácter cambiante de los requisitos, se decide utilizar una metodología ágil; de ellas, se resuelve utilizar AUP para estar en correspondencia con las buenas prácticas definidas en la universidad para el desarrollo de software.

El Proceso Unificado Ágil de Scott Amblar o Agile Inicial Procesos (AUP) es una versión simplificada del Proceso Unificado de Racional (RUP). Esta describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP. Al igual que en RUP, en AUP se establecen tres fases que transcurren de manera consecutiva.

El objetivo de iniciar (Concepción) es establecer el producto visión, plan de proyecto, negocios y justificación tecnológica e identificar los riesgos y las oportunidades. Puede haber uno o más objetivos que deben lograrse en alcanzar este objetivo.

El objetivo de ejecutar (innovación) es evolucionar la visión de producto, plan de proyecto y justificación de negocio y tecnología; identificar y hacer frente a riesgos; identificar y aprovechar oportunidades mientras diseñar, desarrollar (aplicación y pruebas) y desplegar (integración, construcción y libertad) el producto. Normalmente hay muchos objetivos y metas, cada una resultante de la implementación del producto.

El objetivo de cerrar (dejar) es retirar el producto y cerrar el proyecto (talleres de cierre). Puede haber uno o más objetivos que debe lograrse en alcanzar este objetivo.

La visión provee dirección y un enfoque general, esta guía provee dirección específica a través de las metas y las misiones a través de los objetivos, y las colaboraciones establecen intuición, confianza, unidad, y cohesión.

A partir de que el Modelado de negocio propone tres variantes a utilizar en los proyectos (Casos de Uso del Negocio (CUN), Descripción de Proceso de Negocio (DPN) o Mapa Conceptual (MC) y existen tres formas de encapsular los requisitos (Casos de Uso del

Sistema (CUS), Historias de Usuario (HU), Descripción de Requisito por Proceso (DRP)), surgen cuatro escenarios para modelar el sistema en los proyectos, manteniendo en dos de ellos el MC, obteniendo la siguiente forma:

Escenario No 1: Proyectos que modelen el negocio con CUN solo pueden modelar el sistema con CUS.

Escenario No 2: Proyectos que modelen el negocio con MC solo pueden modelar el sistema con CUS.

Escenario No 3: Proyectos que modelen el negocio con DPN solo pueden modelar el sistema con DRP.

Escenario No 4: Proyectos que no modelen negocio solo pueden modelar el sistema con HU.

Como conclusión después de haber revisado los diferentes escenarios dentro de la disciplina de requisitos de software de la metodología AUP, se selecciona el escenario No.4 por ser el más adecuado. Este escenario es ideal para proyectos en los que se haya evaluado claramente el negocio a informatizar y se cuente con una definición sólida del mismo como resultado. En esta situación, el cliente trabajará de cerca con el equipo de desarrollo para negociar los detalles de los requisitos, así como para implementarlos, probarlos y validarlos.

1.5 Arquitectura de software

La arquitectura de software se refiere a una planificación basada en modelos, patrones y abstracciones teóricas, a la hora de realizar una pieza de software de cierta complejidad y como paso previo a cualquier implementación. De esta forma se dispone de una guía teórica detallada que nos permite entender cómo van a encajar cada una de las piezas de nuestro producto o servicio.

Por tanto, en arquitectura llamamos patrón a cualquier solución general y reutilizable para problemas recurrentes en ingeniería del software en un contexto dado, son similares a los patrones usados en la programación, pero orientados específicamente a la estructura a un nivel superior y más genérico.

Generalmente, no es necesario inventar una nueva arquitectura de software para cada sistema de información. Lo habitual es adoptar una arquitectura conocida en función de sus ventajas e inconvenientes para cada caso en concreto. Así, las arquitecturas más universales son:

- Descomposición Modular: El software se estructura en grupos funcionales muy acoplados.
- Cliente-servidor: El software reparte su carga de cómputo en dos partes independientes, pero sin reparto claro de funciones.
- Arquitectura de tres niveles: Especialización de la arquitectura cliente-servidor donde la carga se divide en tres partes (o capas) con un reparto claro de funciones: una capa para la presentación (interfaz de usuario), otra para el cálculo (donde se encuentra modelado el negocio) y otra para el almacenamiento (persistencia). Una capa solamente tiene relación con la siguiente.

Otras arquitecturas menos conocidas son:

- Modelo Vista Controlador.
- En pipeline.
- Entre pares.
- En pizarra.
- Orientada a servicios (SOA del inglés Service-Oriented Architecture).
- Arquitectura de microservicios (MSA del inglés MicroServices Architecture). Algunos consideran que es una especialización de una forma de implementar SOA.
- Dirigida por eventos.[14]

1.5.1 Estilos arquitectónicos

Cada estilo arquitectónico describe una categoría del sistema que contiene: un conjunto de componentes, que realiza una función requerida por el sistema, un conjunto de conectores que posibilitan la comunicación, la coordinación y la cooperación entre los componentes; restricciones que definen como se puede integrar los componentes que forman el sistema; y modelos semánticos que permiten al diseñador entender las propiedades globales de un sistema para analizar las propiedades conocidas de sus partes constituyentes. De estos, se puede mencionar que:

- Sirven para sintetizar estructuras de soluciones.
- Pocos estilos abstractos encapsulan una enorme variedad de configuraciones concretas.
- Definen los patrones posibles de las aplicaciones.
- Permiten evaluar arquitecturas alternativas con ventajas y desventajas conocidas ante diferentes conjuntos de requerimientos no funcionales.

Existen diferentes tipos de estilos arquitectónicos entre los que se encuentran: Centrada en Datos.

- Llamada y retorno.
- Orientada a Objetos.
- Estratificada.[15]

1.6 Tecnologías y herramientas para el desarrollo de la solución

Django (Versión 4.2.2):

Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como modelo–vista–controlador.[16]

Python (Versión 3.12):

Python es un gran **lenguaje de programación para aprender**, y puedes usarlo en una variedad de áreas en el desarrollo de software. Puedes usar Python para desarrollo web, análisis de datos, aprendizaje automático, inteligencia artificial, y más.[17]

SQL (Versión 16.0):

SQL (por sus siglas en inglés Structured Query Language; en español lenguaje de consulta estructurada) es un lenguaje de dominio específico, diseñado para administrar, y recuperar información de sistemas de gestión de bases de datos relacionales.[18]

HTML5:

El Lenguaje de Marcado de Hipertexto (HTML) es el código que se utiliza para estructurar y desplegar una página web y sus contenidos. Por ejemplo, sus contenidos podrían ser párrafos, una lista con viñetas, o imágenes y tablas de datos.[19]

CSS:

Es el lenguaje que se utiliza para controlar el estilo y el diseño de las páginas web.[20]

JavaScript:

Es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web.[21]

PostgreSQL:

Es un sistema de código abierto de administración de bases de datos del tipo relacional, aunque también es posible ejecutar consultas que sean no relaciones. En este sistema, las consultas relacionales se basan en SQL, mientras que las no relacionales hacen uso de JSON.[22]

1.7 Conclusiones parciales

En conclusión, en el capítulo 1 de la tesis se ha presentado una visión general del trabajo propuesto, abordando los aspectos básicos del tema en cuestión. Se ha analizado las características y beneficios de la utilización del tema, lo que ha contribuido al soporte teórico del proyecto. Además, se ha fundamentado el uso del lenguaje de programación y las herramientas utilizadas, lo que permitirá llevar a cabo el desarrollo del proyecto de manera efectiva. Por último, se ha argumentado sobre la metodología de desarrollo seleccionada, lo que permitirá llevar a cabo el proyecto de manera organizada y eficiente. De esta manera, el capítulo 1 sienta las bases teóricas y metodológicas necesarias para el desarrollo del proyecto propuesto.

Capítulo 2: Propuesta de solución

Introducción

En el presente capítulo se describe la propuesta de solución y los artefactos generados por el escenario número 4. Los requisitos funcionales y no funcionales establecen la base para el diseño y desarrollo, definiendo claramente las funcionalidades del sistema. Estos requisitos son la brújula que guía todo el proceso de desarrollo y asegura que el sistema cumpla con las expectativas. Las historias de usuario aportan una comprensión detallada de las necesidades y expectativas de los usuarios finales, ofreciendo una visión holística y centrada en el usuario que influye significativamente en el diseño y la implementación de cada elemento del sistema. La descripción de la arquitectura Modelo-Vista-Plantilla (MVT) es fundamental, ya que proporciona una estructura organizativa que separa los datos, la lógica de negocio y la interfaz de usuario, lo que facilita la mantenibilidad y escalabilidad del sistema. El diagrama de clases de diseño es esencial para comprender la estructura y las interacciones dentro del sistema, proporcionando una representación visual de las entidades y sus relaciones, lo que es crucial para el diseño eficiente y la comprensión del sistema en su totalidad. Respecto a los patrones de diseño, su utilización demuestra un enfoque metodológico y estructurado en la creación de soluciones, lo que puede resultar en un diseño más robusto y con mayor flexibilidad para futuras expansiones y modificaciones. El modelo de datos representa la piedra angular de la gestión de la información, y su diseño cuidadoso es crítico para garantizar la integridad y eficiencia en el almacenamiento y recuperación de los datos. Finalmente, el estándar de codificación empleado es esencial para garantizar la coherencia y legibilidad del código, lo que contribuye al mantenimiento sostenible y a la colaboración efectiva entre desarrolladores.

2.1 Concepción general del sistema

La plataforma web de aprendizaje académico y contenido audiovisual permite la publicación de material didáctico útil para los estudiantes de la Universidad de Cienfuegos. Los usuarios pueden consultar el contenido de forma anónima y reproducir los materiales audiovisuales. Asimismo, tienen la opción de crear un perfil que les permitirá, además de reproducir el contenido, comentar, reaccionar con "me gusta" o "no me gusta", y descargar los materiales publicados. Algunos usuarios desempeñarán el rol

de editores de la plataforma y serán los únicos autorizados para publicar contenido en la misma.

2.1.1 Requisitos Funcionales

Los requisitos funcionales son aquellos que indican lo que debe hacer el producto, son las capacidades con las que debe cumplir el mismo. En la presente investigación se obtuvieron en total 55 requisitos funcionales[23], los cuales se muestran en la siguiente tabla:

No	Nombre	Descripción	Prioridad	Complejidad
1	Listar Películas	El Sistema muestra al usuario las Películas listadas	1	baja
2	Agregar Películas	El Sistema le permite al usuario agregar películas	1	baja
3	Editar información de películas	El Sistema le permite al usuario editar las películas	1	baja
4	Eliminar Películas	El Sistema le permite al usuario eliminar películas	1	baja
5	Listar Series	El Sistema muestra al usuario las Series listadas	2	baja
6	Agregar Series	El Sistema le permite al usuario agregar series	2	baja
7	Editar Series	El Sistema le permite al usuario editar las series	2	baja
8	Eliminar Series	El Sistema le permite al usuario eliminar series	2	baja
9	Listar Temporadas	El Sistema le permite al usuario listar temporadas	3	media

No	Nombre	Descripción	Prioridad	Complejidad
10	Agregar Temporada	El Sistema le permite al usuario agregar las temporadas	3	media
11	Editar información Temporada	El Sistema le permite al usuario editar la información de las temporadas	3	media
12	Eliminar Temporada	El Sistema le permite al usuario eliminar temporadas	3	media
13	Listar Cursos	El Sistema le permite al usuario listar los cursos	3	media
14	Agregar Cursos	El Sistema le permite al usuario agregar cursos	3	media
15	Editar Cursos	El Sistema le permite al usuario editar cursos	3	media
16	Eliminar Cursos	El Sistema le permite al usuario eliminar cursos	3	media
17	Listar Canales	El sistema le permite al usuario listar canales.	3	media
18	Agregar Canal	El Sistema le permite al usuario agregar canales	3	media
19	Editar Canal	El Sistema le permite al usuario editar canales	4	media
20	Eliminar Canal	El Sistema le permite al usuario eliminar canales	4	media

No	Nombre	Descripción	Prioridad	Complejidad
21	Listar Géneros	El Sistema le permite al usuario listar los géneros	4	media
22	Agregar Género	El Sistema le permite al usuario agregar géneros	4	media
23	Editar información del Género	El Sistema le permite al usuario editar la información de los géneros	4	media
24	Eliminar Género	El Sistema le permite al usuario eliminar géneros	4	media
25	Listar actores	El Sistema le permite al usuario listar los actores	4	media
26	Agregar actor	El Sistema le permite al usuario agregar actores.	4	media
27	Editar información de actor	El Sistema le permite al usuario editar la información de los actores.	4	media
28	Eliminar actor	El Sistema le permite al usuario eliminar actores	4	media
29	Listar Almacenamiento Película	El Sistema le permite al usuario listar el	4	media

No	Nombre	Descripción	Prioridad	Complejidad
		almacenamiento de Películas		
30	Agregar Almacenamiento Película	El Sistema le permite al usuario agregar el almacenamiento de Películas	4	media
31	Editar información de Almacenamiento Película.	El Sistema le permite al usuario editar información de Almacenamiento Película.	4	media
32	Eliminar Almacenamiento Película	El Sistema le permite al usuario eliminar el almacenamiento de las películas	4	media
33	Listar Episodios	El Sistema le permite al usuario listar los episodios.	4	media
34	Agregar Episodio	El Sistema le permite al usuario agregar episodios	4	media
35	Editar Episodio	El Sistema le permite al usuario editar los episodios.	4	media
36	Eliminar Episodio	El Sistema le permite al usuario eliminar los episodios.	4	media
37	Listar Almacenamiento Curso	El sistema permite que los usuarios listar el	1	alta

No	Nombre	Descripción	Prioridad	Complejidad
		almacenamiento de los cursos.		
38	Agregar Almacenamiento Curso	El sistema permite que los usuarios puedan agregar almacenamiento de los cursos.	1	alta
39	Editar Almacenamiento Curso	El sistema permite al usuario editar el almacenamiento del curso.	1	alta
40	Eliminar Almacenamiento Curso	El Sistema le permite a los usuarios eliminar almacenamiento curso.	1	alta
41	Loguear usuario	El Sistema permite que los usuarios se logueen.	1	alta
42	Desloguear Usuario	El Sistema permite que los usuarios se deslogueen	1	alta
43	Subscribir Usuario	El Sistema permite que los usuarios se suscriban.	1	baja
44	Listar Usuarios	El Sistema permite listar los usuarios.	3	alta
45	Agregar usuario	El Sistema permite agregar usuarios.	3	alta
46	Agregar Comentario	El Sistema permite al usuario agregar comentarios.	3	alta
47	Responder Comentario	El Sistema permite al usuario responder comentario.	3	alta

No	Nombre	Descripción	Prioridad	Complejidad
48	Descargar Archivo	El Sistema permite al usuario descargar un archivo	2	media
49	Reproducir video	El Sistema permite al usuario reproducir los videos	2	baja
50	Dar me gusta	El sistema permite al usuario dar me gusta	2	baja
51	Dar no me gusta	El sistema permite al usuario dar no me gusta	2	baja
52	Listar calidades	El sistema permite al usuario listar las distintas calidades de video.	2	baja
53	Agregar calidad	El sistema permite al usuario agregar calidades.	2	baja
54	Editar calidad	El sistema permite al usuario editar calidades	2	baja
55	Eliminar calidad	El sistema permite al usuario eliminar calidades.	2	baja

Tabla 1.Requisitos Funcionales

2.1.2 Requisitos no funcionales

- **Requisito de apariencia o interfaz externa**

El Sistema deberá tener una apariencia amigable, su interfaz deberá ser muy legible y debe lucir profesional.

- **Requisito de usabilidad**

La aplicación web debe ser fácil de usar.

- **Requisito de software**

Se requiere de un navegador web instalado en las computadoras clientes.

- **Requisito de Hardware**

El Sistema requiere de una computadora que haga la función de servidor, esta debe cumplir con las siguientes características:

1. Memoria RAM de 1GB o más.
2. Capacidad de disco duro de 50GB o más.

- **Requisito de portabilidad**

La aplicación web podrá ser usada en cualquier computadora, Tablet o teléfono móvil.

- **Requisito de seguridad**

Se debe garantizar un control estricto sobre la seguridad de la información teniendo en cuenta el establecimiento de niveles de acceso. No se deben permitir accesos sin autorización al sistema. Cada usuario podrá interactuar de forma libre por el sistema, al acceder con su usuario y contraseña se le habilitará una serie de características.

- **Requisito de confiabilidad**

Se deberán de tomar las precauciones necesarias para prevenir fallos de cualquier tipo, ya sean por parte de la aplicación o del usuario, previniéndolos mediante validaciones y definiciones a seguir.

2.2 Historias de Usuario

Las historias de usuarios son las técnicas utilizadas para especificar los requisitos del software, son equivalentes a los casos de uso en el proceso unificado y constituyen la base para las pruebas funcionales. En total se definieron 55 historias de usuarios, a continuación, se muestra la historia de usuario de mayor criticidad, “Editar información de las películas”:

Numero: 3	Requisito: Editar información de películas	
Programadores: David Padrón		Iteración asignada: 1

Sheila Pérez	
Prioridad: 1	Tiempo estimado: 10h
Riesgo en desarrollo	Tiempo real
Descripción: El Sistema le permite al usuario editar las películas. Título: Campo de texto donde irá el título de la película. Sinopsis: Campo de texto donde irá de que trata la película. Fecha de estreno: Campo de texto donde irá la fecha en que se estrenó la película con el modo fecha de tipo yyyy/mm/dd. Imagen: Campo de url donde irá la url donde esta almacenada la imagen de la película. Duración: Campo de texto donde irá el tiempo de duración de la película. Director: Campo de texto donde irá el nombre del director de la película. Subtítulos srt: Campo url donde irá la url donde se almacena el subtítulo. Subtítulos vtt: Campo url donde irá la url donde se almacena el subtítulo que contendrá el video de la película en la web.	
Observación: Al editar una película es obligatorio rellenar todos los campos.	
Prototipo elemental de interfaz gráfica de usuario: 	

Tabla 2.Historias de Usuario

2.3 Descripción de la arquitectura

La arquitectura del software constituye una especificación de las principales ideas del diseño proporcionando una descripción más detallada de cómo realizar dicho sistema.

2.3.1 Patrón de arquitectura Modelo Vista Plantilla (MTV)

La propuesta de solución está basada en el patrón Modelo-Vista-Plantilla[24], el cual describe la forma de organizar el código de la aplicación separando los datos de la misma, la interfaz de usuario, y la lógica de control en tres componentes distintos:

- **Modelo:** Componente encargado del acceso a datos, en este componente se encuentran las tablas de la base de datos.
- **Vista:** La vista interactúa con el modelo para obtener los datos. La vista llama a la plantilla.
- **Plantilla:** Esta es la parte que se encarga de mostrar la información con HTML, CSS, etc.

En la siguiente figura, se muestra como interactúan estos tres componentes en la arquitectura definida:

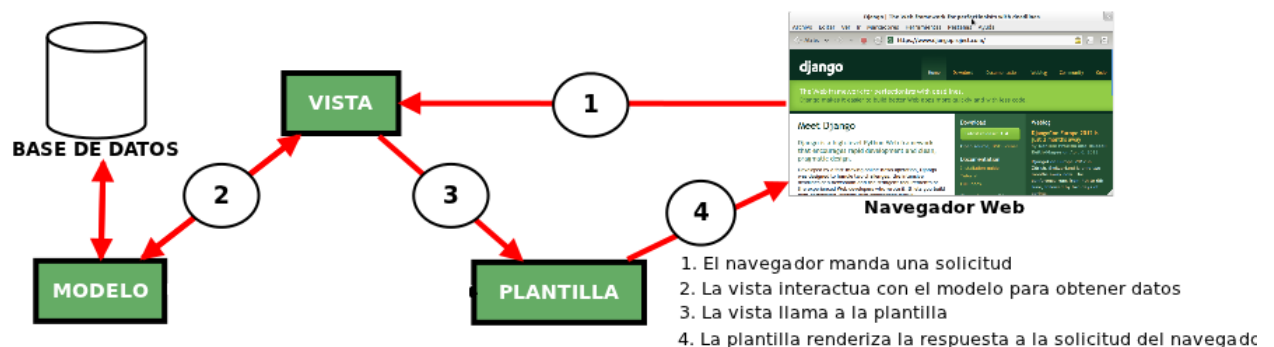


Figura 1. Arquitectura Modelo Vista Template

A continuación, se muestra la aplicación de la arquitectura a la propuesta de solución:

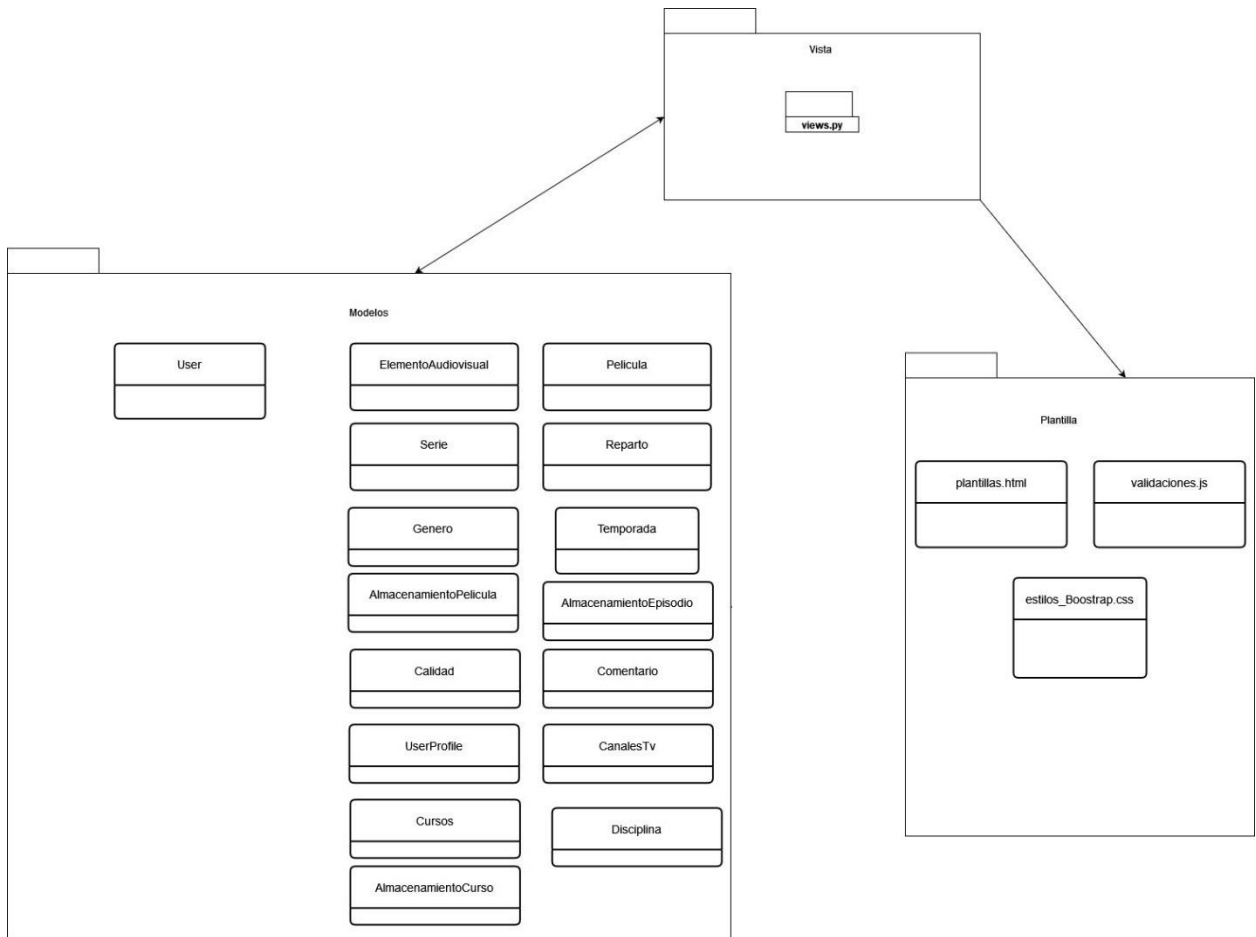


Figura 2. Aplicación de la arquitectura a la solución propuesta

2.4 Diagrama de clases de diseño

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases. Tiene el propósito de describir las clases que conforman el modelo de un determinado sistema. Dado el carácter de refinamiento iterativo que caracteriza un desarrollo orientado a objetos, el diagrama de clase va a ser creado y refinado durante las fases de análisis y diseño, estando presente como guía en la implementación del sistema.

A continuación, se muestra el diagrama de clases de la propuesta de solución:

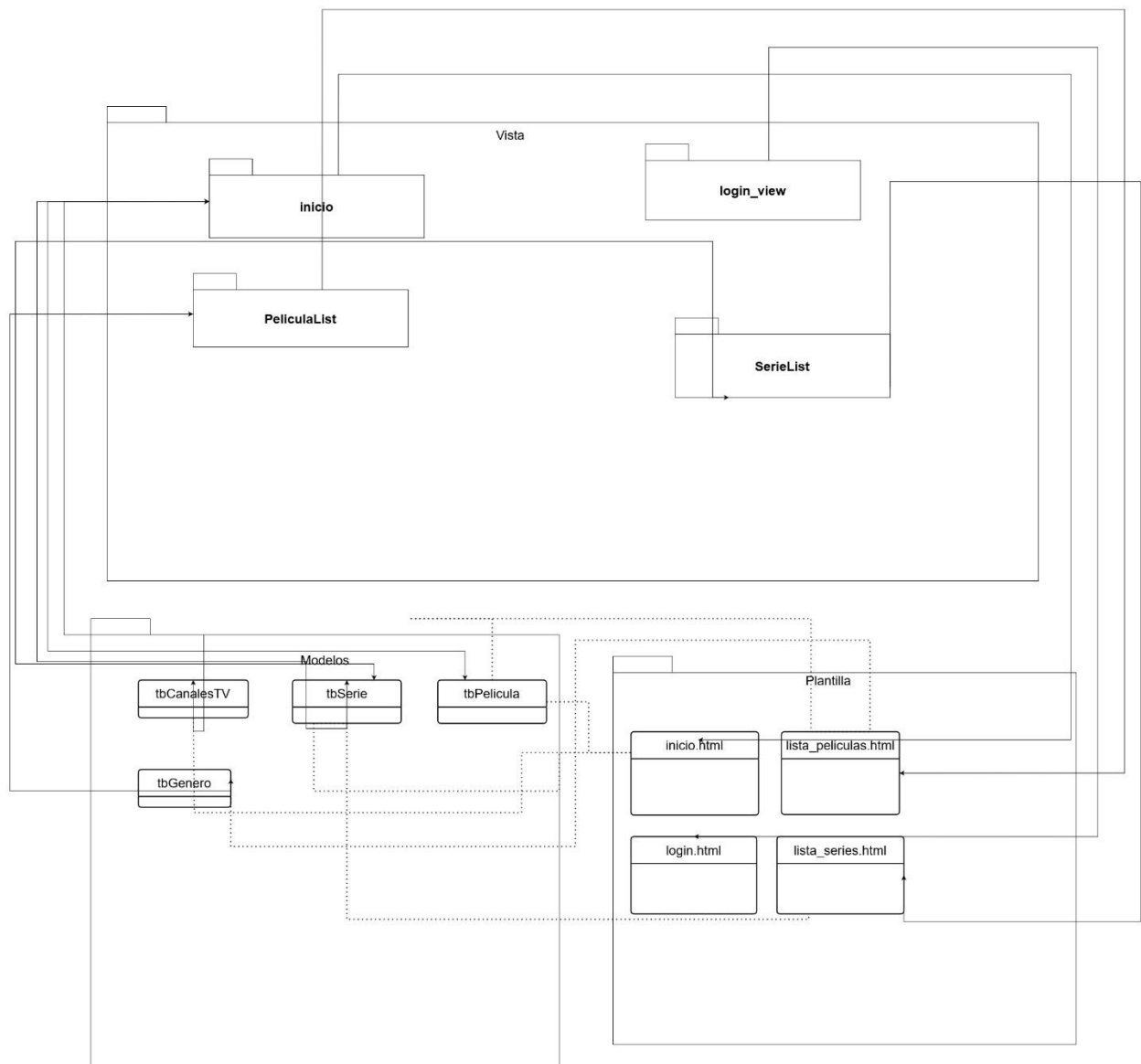


Figura 3. Diagrama de clases del diseño

2.5 Patrones de diseño utilizados

Patrones GRASP: Los patrones GRASP (por sus siglas en inglés, General Responsibility Assignment Software Patterns)[7] describen los principios fundamentales de la asignación de responsabilidades a objetos. El nombre se eligió para indicar la importancia de captar estos principios, si se quiere diseñar eficazmente el software orientado a objetos.

Los patrones GRASP utilizados en la solución propuesta fueron:

Creador: este patrón es quien guía el proceso de asignación de responsabilidades relacionadas con la creación de objetos. Tiene el objetivo de asignarle a la clase B la responsabilidad de crear una instancia de la clase A. Este patrón se pone de manifiesto en la clase Película y otras de las clases utilizadas.

Bajo acoplamiento: Plantea que se debe poder reutilizar las funcionalidades de las distintas clases, con un nivel mínima. Este patrón se evidencia en todas las aplicaciones web que funcionen sobre Django, pues cada pieza de las aplicaciones tiene un propósito clave, que puede modificarse sin afectar otras piezas. Por ejemplo, se puede cambiar la URL de cierta parte de la aplicación sin tener que afectar la implementación subyacente o se puede modificar el HTML de una página sin tener que tocar el código Python que la renderiza.

Alta cohesión: Asigna responsabilidades de manera tal que la cohesión siga siendo alta, o sea que las funcionalidades de las clases estén altamente relacionadas de forma tal que exista una colaboración entre ellas para compartir el esfuerzo y no caiga todo el peso sobre una única clase. Usar este patrón simplifica el mantenimiento y favorece el bajo acoplamiento. Este patrón se utiliza en la clase PeliculaCreateView.

Controlador: Permite manejar todos los eventos del sistema, al servir de intermediario entre las interfaces y el algoritmo que las implementa. Este patrón se evidencia en la clase PeliculaCreateView.

Patrones GOF: Patrones publicados por Gamma, Helm, Johnson y Vlossodes en 1995: patrones de la banda de los cuatro (del inglés, Gang of Four). Esta serie de patrones permiten ampliar el lenguaje, aprender nuevos estilos de diseño y además introducir más notación UML. Existen 23 patrones GoF de los cuales 15 se utilizan con frecuencia.

Los patrones de diseño del grupo GoF se clasifican en tres grandes categorías basadas en su propósito: creacionales, estructurales y de comportamiento. A continuación, se describe el patrón GoF utilizado en la solución propuesta:

Decorador: Patrón estructural que extiende la funcionalidad de un objeto dinámicamente de manera tal que es transparente a sus clientes, utiliza una instancia de una subclase de la clase original que delega las operaciones al objeto original. Este patrón se evidencia en la clase Login require (decorador que trae django por defecto, para acceder a una clase).

2.6 Modelo de Datos

Las bases de datos necesitan una definición de su estructura que le permitan almacenar datos, reconocer el contenido y recuperar la información. La estructura tiene que ser desarrollada para la necesidad de las aplicaciones que la usarán. La puesta en práctica de la base de datos es el paso final en el desarrollo de aplicaciones. Se conforman con los requisitos del proceso del negocio, que es la primera abstracción de la vista de la base de datos.

Un modelo de datos se puede definir como un conjunto de conceptos, reglas y convenciones bien definidos que permiten aplicar una serie de abstracciones a fin de describir y manipular los datos de un cierto mundo real que se desea almacenar en la base de datos. El modelo relacional se caracteriza a muy grandes rasgos por disponer que toda la información debe estar contenida en tablas, y las relaciones entre datos deben ser representadas explícitamente en esos mismos datos. A continuación, se muestra el diseño del esquema de base de datos propuesto a través del modelo de datos:

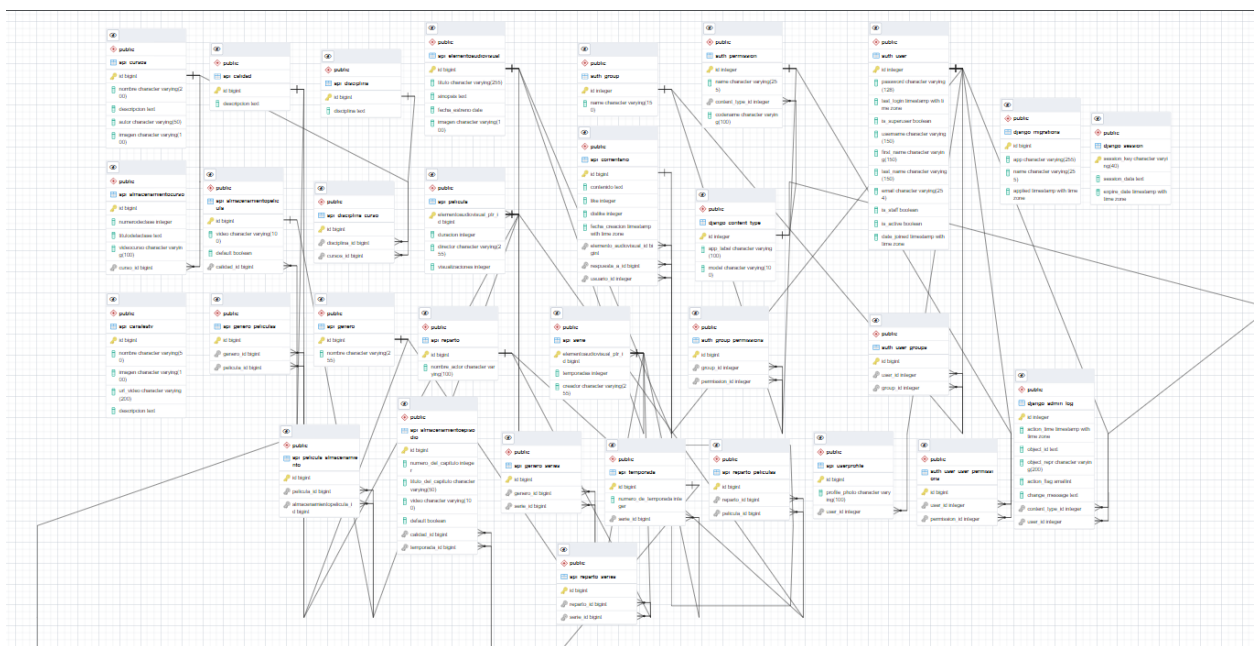


Figura 4. Modelo físico de datos

2.7 Estándar de codificación empleado

Con el objetivo de lograr una estandarización en la programación del sistema se decide aplicar el estándar de codificación CamelCase, el cual facilitará la lectura, comprensión

y mantenimiento del código. A continuación, se describe a grandes rasgos las convenciones de nomenclatura.

Convenciones de nomenclatura General

- Se exceptúan el uso de las tildes y la letra ñ.
- En todo momento se utilizarán nombres que sean claros, concretos y libres de ambigüedades. Ejemplo: "idSeminario" y no solamente "id".
- El nombre de todas las variables y métodos comenzarán con letra minúscula.

Indentación

- El contenido siempre se indentará con *api* para las tablas, nunca utilizando espacios en blanco.

Clases

- El nombre de las clases comenzará con mayúscula y no existirán espacios en blanco.
- Intentar mantener los nombres de las clases descriptivos y simples. Usar palabras completas, evitar acrónimos y abreviaturas, a no ser que la abreviatura sea mucho más conocida que el nombre completo, como URL o HTML.

Nombre de variables

- No se utilizarán nombres de variables que puedan ser ambiguos.
- Se procurará evitar dar nombres sin sentido a variables temporales. Por ejemplo:temp, i, tmp.

2.8 Conclusiones parciales

El sistema diseñado simplifica el proceso de implementación al permitir la ampliación de sus funcionalidades a través de una arquitectura flexible. La aplicación de patrones de diseño garantiza que el sistema sea fácil de entender, extender y mantener, al tiempo que posibilita la reutilización de componentes en futuras aplicaciones. Adicionalmente, el uso de un estándar de codificación facilita la lectura y comprensión del sistema.

Capítulo 3: VALIDACIÓN DE LA VIABILIDAD DE LA PROPUESTA DE SOLUCIÓN

En el presente capítulo se valora la viabilidad de la propuesta de solución mediante la validación de los requisitos y el diseño propuesto, además de la realización de pruebas funcionales, unitarias y de aceptación a la plataforma audiovisual VISUALES UCF.

3.1 Pruebas de software

Niveles de pruebas

Los niveles de prueba organizan y administran actividades para verificar y validar los componentes de un producto. Incluyen prueba de unidad (verifica el funcionamiento de partes individuales), prueba de integración (verifica la interacción entre componentes), prueba de sistema (evalúa el cumplimiento de los requisitos) y prueba de aceptación (verifica la funcionalidad según las necesidades del cliente). Cada nivel utiliza diferentes tipos y técnicas de pruebas para su realización. Para los efectos de la presente investigación se propone la siguiente estrategia de pruebas[26]:

Tipo de Prueba	Método de prueba	Técnica de prueba
Unitarias	Caja blanca	Camino básico
Funcionalidad	Caja negra	Partición de equivalencia

Tabla 3.Estrategia de prueba aplicada a la solución.

3.1.1 Pruebas de caja blanca

La prueba de caja blanca utiliza la estructura de control del diseño procedimental para diseñar casos de prueba. Este enfoque garantiza el ejercicio de todos los caminos independientes de cada módulo, las decisiones lógicas, los bucles en sus límites operacionales y las estructuras internas de datos. Además, se emplea la técnica de prueba del camino básico para asegurar que cada sentencia del programa se pruebe al menos una vez, utilizando la medida de complejidad de un procedimiento o algoritmo para obtener un conjunto básico de caminos de ejecución y casos de prueba[27].

A continuación, se muestra el proceso de pruebas realizado a la funcionalidad login_view, específicamente a la controladora tbUser en donde su funcionalidad consiste

en iniciar sesión en la plataforma. Primeramente, se comienza por analizar el código y enumerar las instrucciones:

```
#Funcion para Loguear mis usuarios
def login_view(request):
    if request.method == 'POST':
        form = AuthenticationForm(request, data=request.POST)
        if form.is_valid():
            username = form.cleaned_data['username']
            password = form.cleaned_data['password']
            user = authenticate(request, username=username, password=password)
            if user is not None:
                login(request, user)
                return redirect(request.GET.get('next', 'inicio'))
            else:
                messages.error(request, 'El usuario o la contraseña son incorrectos.')
                form = AuthenticationForm(request)
        else:
            form = AuthenticationForm(request)
    context = {
        'form': form
    }

    return render(request, 'api/login.html', context)
```

Figura 5.Funcionalidad login_view(loguear usuario).

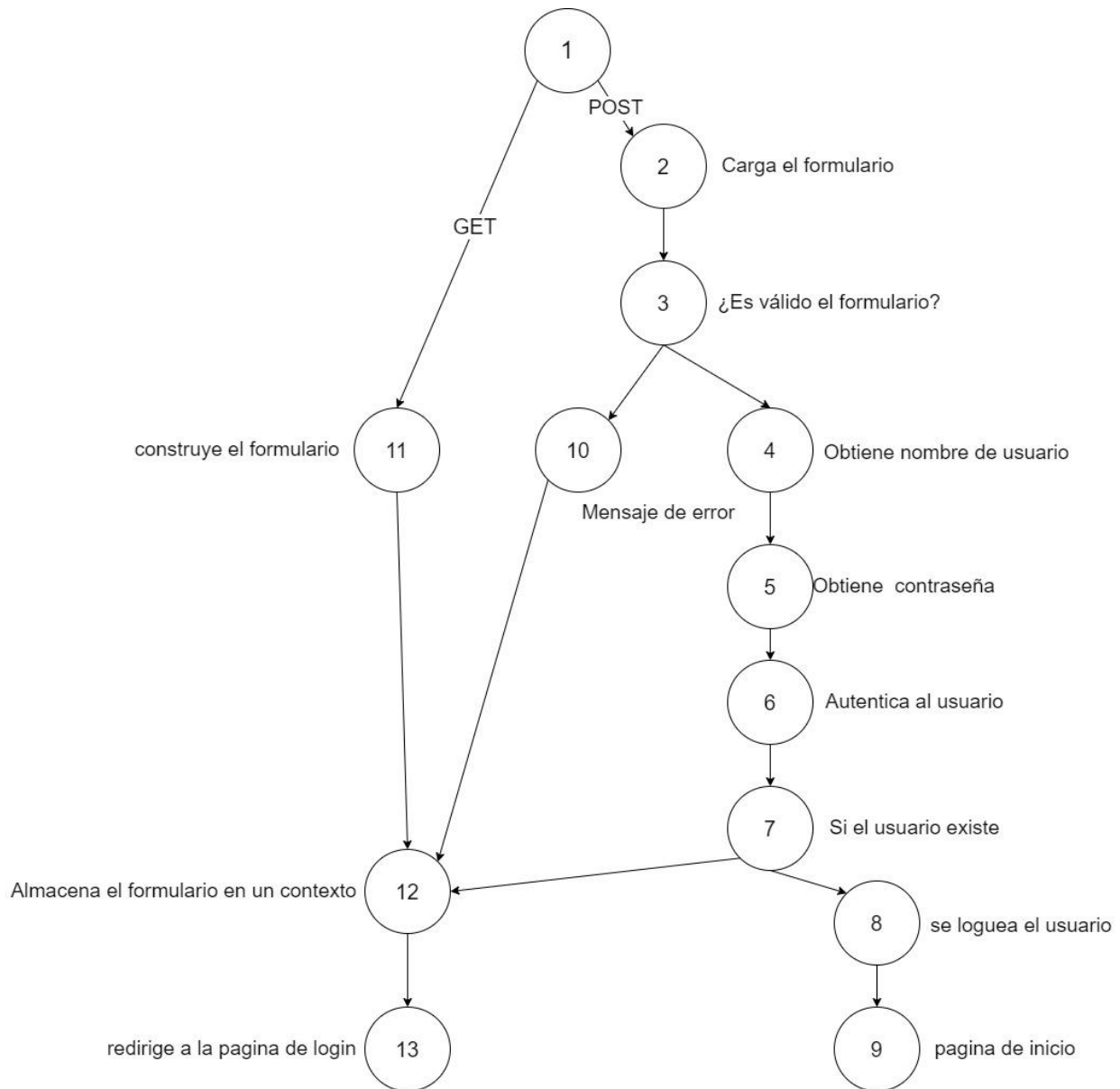


Figura 6 Grafo resultante de la función `login_view`

Luego se determina la complejidad ciclomática $V(G)$ del grafo resultante, la cual es un indicador del número de caminos independientes que existen en un grafo, es decir, es cualquier camino dentro del código que introduce por lo menos un nuevo conjunto de sentencias de proceso o una nueva condición. La complejidad ciclomática puede ser calculada de 3 formas

- $V(G) = a - n + 2$, siendo a el número de arcos o aristas del grafo y n el número de nodos.
- $V(G) = r$, siendo r el número de regiones cerradas del grafo.
- $V(G) = c + 1$, siendo c el número de nodos de condición.

Realizando los cálculos correspondientes se obtiene por cualquiera de las variantes el siguiente resultado:

$$V(G) = a - n + 2$$

$$V(G) = r$$

$$V(G) = c + 1$$

$$V(G) = 14 - 13 + 2$$

$$V(G) = 3$$

$$V(G) = 3$$

$$V(G) = 3$$

Por lo que el conjunto de caminos básico sería:

Camino básico 1: 1-2-3-4-5-6-7-8-9

Camino básico 2: 1-2-3-4-5-6-7-12-13

Camino básico 3: 1-2-3-10-12-13

Camino básico 4: 1-11-12-13

3.1.2 Pruebas de caja negra

Las pruebas de caja negra se enfocan en los requisitos funcionales del software, permitiendo a los ingenieros obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos. Este enfoque es complementario a las técnicas de caja blanca y busca descubrir diferentes tipos de errores, incluyendo funciones incorrectas o incompletas, errores de interfaz, problemas en estructuras de datos o accesos a bases de datos externas, errores de rendimiento, y problemas de inicialización y terminación[28].

Para desarrollar las pruebas de caja negra existen varias técnicas, entre ellas se encuentran:

- **Técnica de la Partición de Equivalencia:** esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- **Técnica del Análisis de Valores Límites:** esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables
- **Técnica de Grafos de Causa-Efecto:** es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

La técnica de Partición de equivalencia es una técnica de prueba que se utiliza para reducir el número de casos de prueba necesarios para probar un rango de valores de entrada[29]. Para aplicar la prueba de partición de equivalencia a un caso de uso Registrar usuario, podríamos dividir cada campo en rangos válidos e inválidos.

Identificación de particiones de equivalencia:

- Nombre de usuario:
 - Partición válida: Nombre de usuario no vacío.
 - Partición no válida: Nombre de usuario vacío.
- Correo electrónico:
 - Partición válida: Correo electrónico con formato válido, que no existe previamente en la base de datos.
 - Partición no válida: Correo electrónico con formato inválido, que ya existe en la base de datos.
- Contraseña:
 - Partición válida: Contraseña con al menos 8 caracteres.
 - Partición no válida: Contraseña con menos de 8 caracteres.
- Confirmar contraseña:
 - Partición válida: Confirmación de contraseña que coincide con la contraseña.

- Partición no válida: Confirmación de contraseña que no coincide con la contraseña.

Casos de prueba:

- Nombre de usuario:

- Caso de prueba 1 (Válido): Nombre de usuario "usuario123".

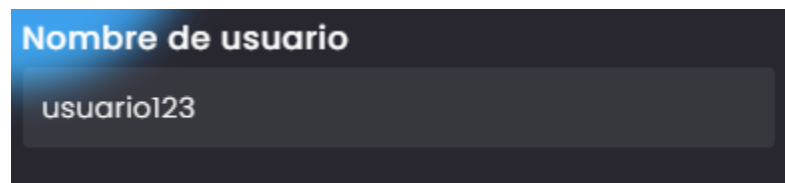
A screenshot of a login form with a dark background. At the top, the text "Nombre de usuario" is displayed in white. Below it is a text input field containing the text "usuario123".

Figura 7 Caso de prueba 1

- Caso de prueba 2 (No válido): Nombre de usuario vacío.

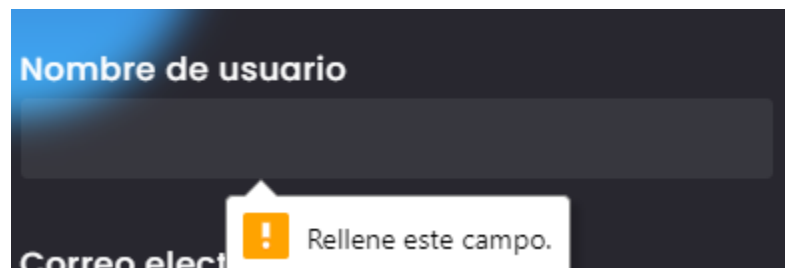
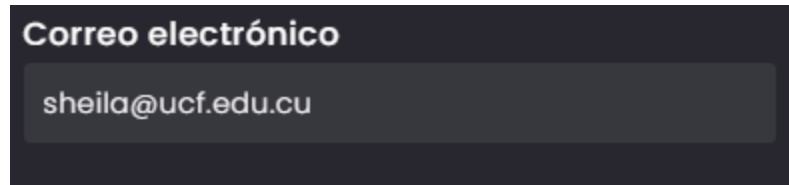
A screenshot of a login form with a dark background. At the top, the text "Nombre de usuario" is displayed in white. Below it is an empty text input field. A yellow error message box with an exclamation mark icon is visible, containing the text "Rellene este campo.".

Figura 8 Caso de prueba 2

- Correo Electrónico:

- Caso de prueba 3 (Válido): Correo electrónico válido que no está registrado en la base de datos.

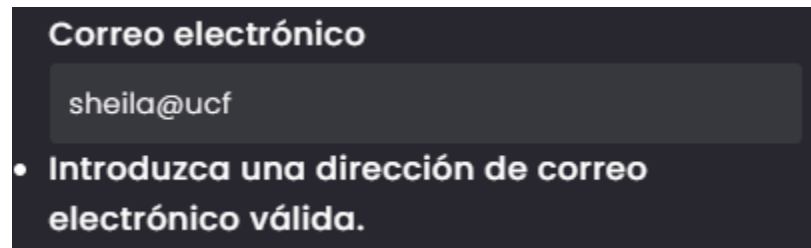


Correo electrónico

sheila@ucf.edu.cu

Figura 9 Caso de prueba 3

- Caso de prueba 4 (No válido): Correo electrónico con formato inválido.



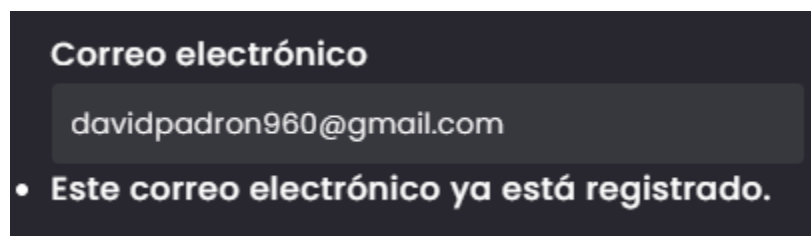
Correo electrónico

sheila@ucf

- Introduzca una dirección de correo electrónico válida.

Figura 10 Caso de prueba 4

- Caso de prueba 5 (No válido): Correo electrónico que ya está registrado en la base de datos.



Correo electrónico

davidpadron960@gmail.com

- Este correo electrónico ya está registrado.

Figura 11 Caso de prueba 5

- Contraseña:

- Caso de prueba 6 (Válido): Contraseña "password123".



Contraseña

.....

Figura 12 Caso de prueba 6

- Caso de prueba 7 (No válido): Contraseña "pass".

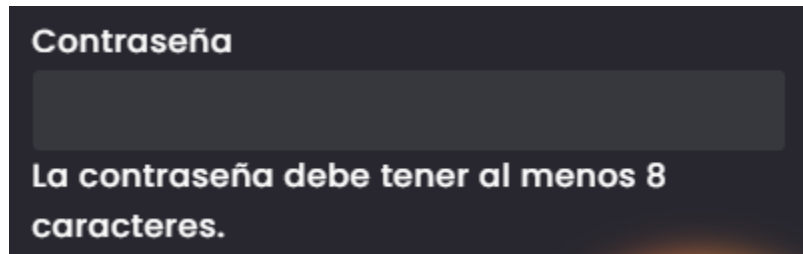


Figura 13 Caso de prueba 7

- Confirmar Contraseña:
 - Caso de prueba 8 (Válido): Confirmación de contraseña que coincide con la contraseña.
 - Caso de prueba 9 (No válido): Confirmación de contraseña que no coincide con la contraseña.

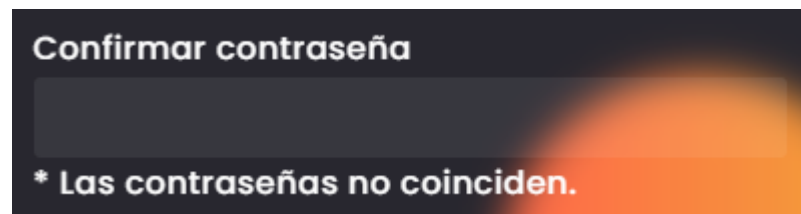


Figura 14 Caso de prueba 9

3.2 Diseño de pruebas funcionales

Las pruebas son un aspecto crucial en el control de calidad del desarrollo de software y, dentro de estas, las pruebas funcionales, en las cuales se verifica de forma dinámica el comportamiento de un sistema, basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba. Las pruebas funcionales son las que se aplican al producto final, permitiendo detectar los puntos del producto que no cumplen sus especificaciones, es decir, que no funcionan correctamente[30].

Caso de uso	Agregar películas
-------------	-------------------

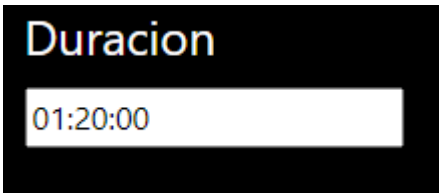
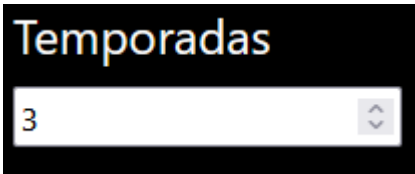
Resumen	El Sistema le permite al usuario agregar películas
Pruebas funcionales del sistema	
Sección para insertar los datos de la película.	
Es la interfaz donde el usuario crea las películas que va a tener la plataforma.	
 Validación : El framework Django se encarga de hacer las validaciones mediante formularios donde se crean reglas para controlar el tipo de datos que se ingresen en los campos, en este caso la duración debe contener solo números. Si la validación no tuvo éxito, los campos donde ocurrió el error aparecerán con una pequeña descripción del error. 	

Tabla 4. Prueba funcional para el caso de uso Agregar Películas.

Caso de uso	Agregar serie
Resumen	El Sistema le permite al usuario agregar serie
Pruebas funcionales del sistema	
Sección para insertar los datos de la serie.	
Es la interfaz donde el usuario crea las series que va a tener la plataforma.	
 Validación :	

El framework Django se encarga de hacer las validaciones mediante formularios donde se crean reglas para controlar el tipo de datos que se ingresen en los campos, en este caso la duración debe contener solo números.

Si la validación no tuvo éxito, los campos donde ocurrió el error aparecerán con una pequeña descripción del error.

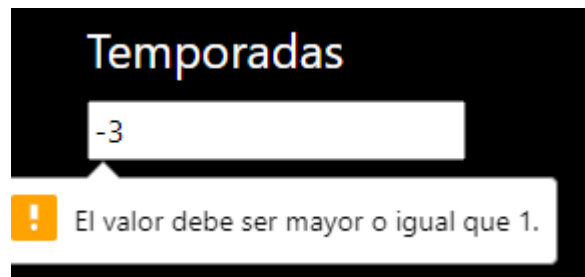


Tabla 5. Prueba funcional para el caso de uso Agregar Serie.

3.3 Conclusiones parciales

Las pruebas de Caja Negra han verificado que las funciones del software operan eficazmente a través de la interfaz, garantizando una correcta aceptación de la entrada y la generación de resultados precisos. Estas pruebas garantizan la integridad de la información externa. Por otro lado, las pruebas de Caja Blanca han servido para validar internamente las funciones de la plataforma, facilitando la detección de posibles desviaciones para su pronta corrección. Además, las pruebas son esenciales en el control de calidad del desarrollo de software. En particular, las pruebas funcionales, las cuales dinámicamente validan el comportamiento del sistema y se basan en la observación de un conjunto selecto de ejecuciones controladas o casos de prueba. Estas pruebas se aplican al producto final, permitiendo detectar desviaciones en relación a sus especificaciones funcionales, es decir, identificando áreas en las que el producto no opera correctamente.

Conclusiones generales

Durante el desarrollo de este proyecto de diploma, se logró la creación exitosa de una plataforma de aprendizaje estudiantil con enfoque en contenido audiovisual y aprendizaje académico para la Universidad Carlos Rafael Rodríguez. Este trabajo implicó:

- Identificación detallada de las necesidades y preferencias de los usuarios de la plataforma, lo que resulta fundamental para el diseño de futuras mejoras y la creación de experiencias más personalizadas, contribuyendo así a incrementar la satisfacción y el compromiso de los usuarios.
- La elección de la metodología AUP facilitó la creación de elementos de ingeniería como historias de usuario, el patrón arquitectónico MVT, diagramas de clases y modelos de datos, que sirvieron como guías para la implementación.
- El empleo de herramientas y tecnologías como UML 2.0 mediante la herramienta Draw 22.1.2, el lenguaje de programación Python 3.12 con su Framework Django 4.2.2, el entorno de desarrollo Visual Studio Code 1.84.2 y PostgreSQL como sistema de gestión de base de datos, fue crucial para establecer la infraestructura tecnológica necesaria para el desarrollo de la solución propuesta.
- La implementación de la plataforma audiovisual para el entretenimiento y el aprendizaje estudiantil garantiza la estructuración, consistencia y persistencia de la información de los elementos audiovisuales gestionados en la misma.
- Las evaluaciones realizadas, junto con la implementación de técnicas de validación, confirmaron la satisfacción por parte de los usuarios, lo cual contribuyó significativamente a la calidad de la plataforma, reforzando así la organización, coherencia y durabilidad de la información dentro del entorno de la plataforma.

Recomendaciones

Para dar continuidad a la presente investigación se recomienda:

- Continuar el estudio con el objetivo de añadir nuevas funcionalidades al sistema.
- Desarrollar una aplicación para dispositivos móviles que permita el acceso a la información audiovisual desplegada en la plataforma con similares funcionalidades a la versión web.

Referencias

- [1] D. J. Bermejo, «Tecnologías de la información y comunicación (TIC) - Definición, qué es y concepto», Economipedia. Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://economipedia.com/definiciones/tecnologias-de-la-informacion-y-comunicacion-tic.html>
- [2] «Cuba y el impacto de las TIC en la informatización de la sociedad». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://www.monografias.com/trabajos109/cuba-y-impacto-tic-informatizacion-sociedad/cuba-y-impacto-tic-informatizacion-sociedad>
- [3] «mision», Universidad de Cienfuegos. Accedido: 20 de octubre de 2023. [En línea]. Disponible en: https://www.ucf.edu.cu/?page_id=407
- [4] «Construcción de contenidos educativos digitales como apoyo a los procesos de aprendizaje en la Universidad Dr. José Matías Delgado». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://recursos.educoas.org/publicaciones/construccion-de-contenidos-educativos-digitales-como-apoyo-los-procesos-de-aprendizaje>
- [5] «Qué es una plataforma digital, qué tipos existen y ejemplos». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://blog.hubspot.es/website/que-es-plataforma-digital>
- [6] «Audiovisual», *Wikipedia, la enciclopedia libre*. 1 de febrero de 2023. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=Audiovisual&oldid=148988465>
- [7] «Medios de Enseñanza - EcuRed». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: https://www.ecured.cu/Medios_de_Ense%C3%B1anza
- [8] L. M. de la Torre Navarro y J. Domínguez Gómez, «Las TIC en el proceso de enseñanza aprendizaje a través de los objetos de aprendizaje», *Rev. Cuba. Informática Médica*, vol. 4, n.º 1, pp. 83-92, jun. 2012.
- [9] «¿Qué es Platzi? La apuesta para que nunca pares de aprender», Platzi. Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://platzi.com/blog/que-es-platzi/>
- [10] «Udemy», *Wikipedia, la enciclopedia libre*. 9 de junio de 2022. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=Udemy&oldid=144082090>
- [11] «¿Qué es Netflix?» Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://help.netflix.com/es/node/412>

- [12] «Plex Pass: qué es y para qué sirve». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://www.xataka.com/basics/plex-pass-que-sirve>
- [13] «Picta», *Wikipedia, la enciclopedia libre*. 9 de octubre de 2023. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=Picta&oldid=154437463>
- [14] E. G. Maida y J. Pacienza, «Metodologías de desarrollo de software», 2015, Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://repositorio.uca.edu.ar/handle/123456789/522>
- [15] «Arquitectura de software: Qué es y qué tipos existen | OpenWebinars». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://openwebinars.net/blog/arquitectura-de-software-que-es-y-que-tipos-existen/>
- [16] «Estilos arquitectónicos - EcuRed». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: https://www.ecured.cu/Estilos_arquitect%C3%B3nicos
- [17] «Django (framework)», *Wikipedia, la enciclopedia libre*. 8 de octubre de 2023. [En línea]. Disponible en: [https://es.wikipedia.org/w/index.php?title=Django_\(framework\)&oldid=154430337](https://es.wikipedia.org/w/index.php?title=Django_(framework)&oldid=154430337)
- [18] «¿Qué es Python? | Blog Becas Santander». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://www.becas-santander.com/es/blog/python-que-es.html>
- [19] «SQL», *Wikipedia, la enciclopedia libre*. 30 de septiembre de 2023. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=SQL&oldid=154184430>
- [20] «Conceptos básicos de HTML - Aprende desarrollo web | MDN». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics
- [21] D. Santos, «Introducción al CSS: qué es, para qué sirve y otras 10 preguntas frecuentes». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://blog.hubspot.es/website/que-es-css>
- [22] «¿Qué es JavaScript? - Explicación de JavaScript (JS) - AWS». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://aws.amazon.com/es/what-is/javascript/>
- [23] «¿Qué es PostgreSQL? - Para qué sirve, Características e Instalación». Accedido: 20 de octubre de 2023. [En línea]. Disponible en: <https://blog.infranetworking.com/servidor-postgresql/>

- [24] R. S. Pressman, *Ingeniería de Software - Un enfoque práctico*, 7th ed. New York: McGraw-Hill Higher Education, 2010.
- [25] A. Holovaty y J. Kaplan-Moss, *The definitive guide to Django: Web development done right*, 2nd ed. en *The experts's voice in Web development*. Berkeley, CA : New York: Apress ; Distributed to the book trade worldwide by Springer-Verlag, 2009.
- [26] C. Larman, *Applying UML and patterns: an introduction to object-oriented analysis and design and the unified process*, 2nd ed. Upper Saddle River, NJ: Prentice Hall PTR, 2002.
- [27] F. Sánchez, «Tipos de Pruebas en Desarrollo de Software», Troop Software Factory. Accedido: 5 de diciembre de 2023. [En línea]. Disponible en: <https://www.cmsparamedios.com.ar/tipos-de-pruebas-en-desarrollo-de-software/>
- [28] «Pruebas de caja blanca - EcuRed». Accedido: 26 de noviembre de 2023. [En línea]. Disponible en: https://www.ecured.cu/Pruebas_de_caja_blanca
- [29] «Pruebas de caja negra - EcuRed». Accedido: 26 de noviembre de 2023. [En línea]. Disponible en: https://www.ecured.cu/Pruebas_de_caja_negra
- [30] C. Gomez, «Técnicas de prueba de caja negra», Diario de QA. Accedido: 26 de noviembre de 2023. [En línea]. Disponible en: <https://www.diariodeqa.com/post/tecnicas-de-prueba-de-caja-negra>
- [31] L. Gonz´y L. Palacio, «Metodo para generar casos de prueba funcional en el desarrollo de software.», *Rev. Ing.*, pp. 29-37, jul. 2009.