APLICATIVO WEB Y APP MÓVIL QUE SISTEMATICE LAS REGLAS DE NEGOCIO EN UNA BIBLIOTECA DE UNA INSTITUCIÓN EDUCATIVA MEDIANTE LA ARQUITECTURA ORIENTADA A LOS MICROSERVICIOS.

**Autores:**

CERÓN PORTILLA BRAYAN DANIEL, brayan0n@udenar.edu.co

Propuesta de trabajo de grado presentado como requisito parcial para optar al título de Ingeniero de Sistemas, en modalidad investigación.

PROGRAMA DE INGENIERÍA DE SISTEMAS

DEPARTAMENTO DE SISTEMAS

FACULTAD DE INGENIERÍA

UNIVERSIDAD DE NARIÑO

OCTUBRE, 2022

# TABLA DE CONTENIDO

Pág.

[TABLA DE CONTENIDO 2](#_Toc121520367)

[INTRODUCCIÓN 3](#_Toc121520368)

[I. TÍTULO 4](#_Toc121520369)

[II. GRUPO Y LÍNEA DE INVESTIGACIÓN 5](#_Toc121520370)

[III. RESUMEN DEL PROYECTO 6](#_Toc121520371)

[IV. PLANTEAMIENTO DEL PROBLEMA 7](#_Toc121520372)

[A. DESCRIPCIÓN DEL PROBLEMA 7](#_Toc121520373)

[B. FORMULACIÓN DEL PROBLEMA 8](#_Toc121520374)

[V. JUSTIFICACIÓN 9](#_Toc121520375)

[VI. ANTECEDENTES 12](#_Toc121520376)

[VII. OBJETIVOS 17](#_Toc121520377)

[A. OBJETIVO GENERAL 17](#_Toc121520378)

[B. OBJETIVOS ESPECÍFICOS 17](#_Toc121520379)

[VIII. MARCO TEORICO 18](#_Toc121520386)

IX. RESULTADOS ......................................................................................................................30

[Bibliografía 38](#_Toc121520387)

[ANEXOS 43](#_Toc121520388)

# INTRODUCCIÓN

El aprendizaje es vital en la calidad de vida del ser humano, se parte desde un aprendizaje autónomo y otro por medio de enseñanza como se logra en las instituciones educativas, de acuerdo con esto en las instituciones educativas existen muchas áreas y lugares de aprendizaje y una de ellas es la biblioteca, donde los estudiantes van a adquirir conocimiento ya sea por cumplir con sus asignaturas o de otra forma consultar libros de los temas en los cuales estén interesados.

Por otra parte, usar las tecnologías de la información se vuelve un factor clave en el desarrollo ya que en un mundo actual cada día más avanzado tecnológicamente se requiere hacer uso de los sistemas desarrollados ya sea para control y gestión de la información, así mismo los sistemas avanzan cada vez más y hay que adaptarse a estos nuevos y mejorados desarrollos.

Partiendo de ello, se plantea el desarrollo de un aplicativo web y aplicativo móvil que sistematice las reglas de negocio de la biblioteca en una institución educativa basado en la arquitectura de microservicios, lo que busca en sí, es tener un software el cual haga el control de la biblioteca partiendo de un login al sistema y en adelante el registro de libros y el control del inventario y entre otras cosas y que dicho software se encuentre actualizado, además que permita ser escalable siendo competitivo con los requerimientos del mercado actual.

Según lo anterior esta propuesta se organiza de tal manera en que en un comienzo se centra en describir lo que se va a utilizar para el desarrollo, en un segundo apartado tenemos el desarrollo en sí, de las apps móvil y web y luego se pone en marcha mediante un caso de uso que sería básicamente llevarlo a un entorno real en el cual se implemente este software en una institución educativa, y se logre evidenciar que se cumplen con los requerimientos planteados y que contribuye al buen desarrollo del aprendizaje en la institución educativa.

Finalmente se cuenta con unos antecedentes que nos permiten de alguna manera guiar el camino a seguir, luego de ello se plasman los objetivos en una metodología y para finalizar se establece un cronograma de actividades a seguir y un estimado del presupuesto.

# TÍTULO

Aplicativo web y app móvil que sistematice las reglas de negocio en una biblioteca de una institución educativa mediante la arquitectura orientada a los microservicios.

# GRUPO Y LÍNEA DE INVESTIGACIÓN

Grupo: GRIAS

Línea de investigación:

-Desarrollo de software

-Descubrimiento de conocimiento de bases de datos.

-Inteligencia de negocios.

# RESUMEN DEL PROYECTO

Esta propuesta nace a partir de una problemática en la cual se cuenta con sistemas de información y control obsoletos y poco fiables con arquitecturas obsoletas, por consiguiente, se quiere obtener como objetivo principal un producto que permita la mejora y sistematización de las reglas de negocio en una biblioteca de una institución educativa.

Junto con ello, el desarrollo y tecnologías de la información se encuentran en constante cambio, un claro ejemplo son las arquitecturas de desarrollo de software, al centrarse en un entorno educativo como lo es el desarrollo de software para una institución educativa, el resultado que se obtenga será de gran ayuda para la gestión y control de la biblioteca, y encaminándose a unos objetivos específicos como son la caracterización y descripción de los procesos a llevar.

Posteriormente se pretende lograr el desarrollo antes mencionado, mostrado en un aplicativo web y aplicativo móvil que contenga un sistema de gestión y control para una biblioteca de una institución, esto desarrollado en metodologías ágiles las cuales permiten el buen control de desarrollo tanto back-end como front-end y basados en los microservicios que, si bien hoy no son tan comunes, es un nuevo paso hacia la actualidad que permiten la mejora continua y la escalabilidad de los sistemas.

Finalmente, el producto desarrollado se incluye en un caso de uso como lo puede ser una biblioteca de una institución, luego de ello se harán las debidas pruebas y se analizará la experiencia de usuario que tiene el producto final y su aceptación con los consumidores finales, en este caso particular los estudiantes y encargados de la biblioteca.

**Palabras clave:** Arquitectura de microservicios, back-end, front-end, reglas de negocio, desarrollo actual.

# PLANTEAMIENTO DEL PROBLEMA

## DESCRIPCIÓN DEL PROBLEMA

El desarrollo escolar es parte fundamental en la vida, ya que es el paso a un conocimiento muy amplio, de acuerdo con el gobierno nacional en el código de infancia y adolescencia en su Artículo 28 señala que “los niños, y las niñas y los adolescentes tienen derecho a una educación de calidad. Esto será obligatorio por parte del Estado en un año de preescolar y nueve de educación básica”[1], partiendo con esto se da lugar a hablar del software de las instituciones educativas y enfocándonos en el proceso que se lleva con las bibliotecas escolares ya que es un pilar muy importante en el desarrollo escolar.

En las instituciones educativas, a menudo es notable ver como existe un software basado ya sea en aplicaciones instalables, apps web o apps móviles que de algún modo están fabricados en arquitecturas que no son modernas [2], esto da lugar a que existan inconvenientes y problemas específicos en la institución, de tal manera se podría evitar si dicho software o apps están en constante mejoramiento y actualizaciones por parte de los desarrolladores. En caso contrario se ve presente el uso de software y sistemas web que están obsoletos y no tienen soporte por parte de los desarrolladores [3], esto será siempre un factor limitante ya que no se hace uso de aplicaciones web especialmente diseñadas para cada caso, y solo se limitan a las funciones genéricas como por ejemplo prestar libro, devolver libro, que tiene un conjunto determinado de instituciones en funcionamiento.

Por otro lado, se cuenta con las instituciones las cuales hacen uso de su propio software o de algún modo tienen personal encargado en el desarrollo web así que al ser menos profesional se llegan a tener inconvenientes en cuanto a la ejecución de los sistemas [4], dado el caso se encuentra con páginas web poco optimizadas, apps de instituciones que están diseñadas bajo métodos de desarrollo poco fiables u obsoletos.

Los problemas que se presentan con el software y apps actuales es la poca escalabilidad que se presenta, ya que existen arquitecturas basadas en desarrollo como la monolítica en la cual todo el desarrollo se centra en un solo sistema y esto da paso a fallos totales en el sistema si una parte del mismo está en fallo [5], como también la arquitectura de servicios si bien, es funcional parte de módulos grandes los cuales tienen en su interior muchas funciones [6], y aunque es viable de algún modo, sí falla un servicio va estar también fallando los componentes que hagan parte de él.

Finalmente se toma en cuenta la problemática que se observa con las reglas de negocio, hay un factor común existente que es la deficiente administración que se tiene con los sistemas web y los servicios en las tecnologías de la información, de tal modo que al no tener bien definidas las reglas de negocio se hace complejo el uso de estos sistemas.

## FORMULACIÓN DEL PROBLEMA

¿Cómo sistematizar las reglas de negocio de una biblioteca en una institución educativa usando la arquitectura de microservicios?

# JUSTIFICACIÓN

El mundo actual está en constante cambio, y más si se habla del desarrollo tecnológico, hoy en día se cuenta con avances informáticos y tecnologías que hace unas cuantas décadas eran algo imposible o poco probable, de este modo se toman en cuenta la construcción y desarrollo de software que, si bien se ha hecho desde hace mucho, hay que tomar en cuenta que también presenta cambios y mejoras cada día.

Así mismo, el desarrollo que se hace en software tiene partes importantes entre ellas una que se aborda en el presente es la arquitectura que se maneja, se cuenta con arquitecturas que han funcionado bien de alguna manera en el pasado como ya se habló de la arquitectura monolítica o la arquitectura de servicios, pero también con el pasar de los años van quedando obsoletas y no actualizadas [2],es aquí donde se comienza a estudiar lo nuevo y a su vez mejorado como lo es la arquitectura de microservicios.

Por otro lado, se hace el desarrollo app web y app móvil con el uso de los microservicios, los microservicios son un enfoque arquitectónico y organizativo para el desarrollo de software donde el software está compuesto por pequeños servicios independientes que se comunican a través de API bien definidas [7],además los propietarios de estos servicios son equipos pequeños independientes. La arquitectura de microservicios hace que las aplicaciones sean más fáciles de escalar en un entorno en el cual se mantenga estar actualizado con las nuevas tecnologías, así como también esta arquitectura es más rápida de desarrollar porque al momento de desarrollar se pueden dividir a los desarrolladores cada parte de la aplicación web y al momento de crear el microservicio tendrá su propio apartado el cual funciona independientemente del resto.

Lo anterior descrito, permite la innovación ya que se toma en cuenta las nuevas tecnologías de desarrollo así que se va tener puntos a favor en cuanto a mejoras de software de igual manera acelera el tiempo de comercialización de las nuevas características, si bien, al hacer el desarrollo a una institución educativa esto no implica que no se pueda comercializar ya que, si se tiene la base de desarrollo se puede implementar en diferentes organizaciones, adaptando las necesidades y requerimientos que se soliciten.

A su vez, es importante rescatar que en una arquitectura de microservicios se toma en cuenta el desarrollo web y software ya dispuesto en la institución educativa y de algún modo se hace una migración a este modelo de arquitectura de microservicios que básicamente conlleva tener apartados para cada función en el proceso, esto básicamente se resume en utilizar el software ya dispuesto, si de alguna manera existe y está disponible, y tratar de plasmarlo con la arquitectura que se está implementando.

Además con el uso de esta arquitectura de microservicios se da un gran paso en cuanto a la complejidad que exige el manejo de grandes flujos de información y la necesidad de mantener sincronizados y actualizados los datos [8], según el caso actual que es en una biblioteca de una institución educativa, ya que existen colecciones que se están actualizando, y la base de datos requiere que el servicio que lo consume sea rápido, eso se puede comparar con las grandes empresas en la cuales la búsqueda en algún apartado se hace de manera muy rápida y eficaz, esto se logra gracias al uso de la intranet que hace parte de la institución.

Cabe destacar que el manejo adecuado de los microservicios ayuda con la actualización de web obsoletas así como también permite independientemente de las tecnologías que vayan surgiendo en el camino adaptarse fácilmente, en un microservicio se toma en cuenta el uso, que tiene como ventaja que está compuesto por sus diferentes componentes, que permiten que esté en funcionamiento y de alguno modo si falla entonces solo fallara dicho apartado ya que funcionan de manera aislada trabajando de manera separada con los demás apartados, esto permite que se hable de seguridad, actualmente según López,[7]“La seguridad es un problema importante cuando se trata de la arquitectura de microservicios y los sistemas distribuidos en general. En microservicios, garantizar la seguridad se vuelve mucho más compleja por eso en el desarrollo que se haga se debe garantizar óptimas condiciones y no deben existir fallos en seguridad ni tampoco vulnerabilidades esto se logra llevando buenas prácticas de desarrollo y tener alguien ajeno al equipo de desarrollo haciendo testeos y buscando fallas en el sistema para así ser corregidas.

De acuerdo con esto se logra que el software de las instituciones se modernice para estar siendo competitivo con el mercado, de igual modo que dicho software permita que se utilice para generar valor en la institución puesto que es un gran aporte en cuanto al desarrollo, igualmente se busca tener un software seguro, ya que otras arquitecturas como la monolítica presenta fallos a nivel de tener todos los procesos juntos de tal manera que si falla algo en este software por ende fallara todo el sistema [9],al mismo tiempo el software se hace fácil de actualizar, contrario a cuando no se tiene implementada la arquitectura de microservicios es complejo de actualizar el equipo de desarrollo ya que necesita mejores técnicas, actualizaciones completas del sistema y esto se resume en bajo rendimiento, disponibilidad, que no se ve presente y se pretende lograr.

Finalmente se complementa con las reglas de negocio de una institución educativa que conforman una parte muy importante, con esto se permite llevar a cabo una sistematización de las reglas de negocio que, en sí, hace referencia a tratar de mejorar las practicas que se llevan a cabo dentro de la institución y esto nos permitirá organizar cada subproceso que se realiza dentro de una biblioteca y con esto se pretende que el sistema y sus reglas queden acordes con el desarrollo web fomentando un buen uso del software.

# ANTECEDENTES

Arias, Echeverri y Hernández (2021) en su estudio **“**Sistema de información para la medición del riesgo de liquidez en el sector solidario colombiano.” se tiene como objetivo el desarrollo de un sistema web, que permite cuantificar el valor del IRL usando una construcción arquitectónica del mismo orientada por microservicios, obteniendo como resultado que se evidencia que para el periodo en estudio (año 2020), en la primera banda se cuenta con valores del IRL promedio del 2.29, lo que indica que la entidad cuenta con más del doble de ingresos para cubrir sus egresos, lo que de acuerdo con las escalas definidas en la parametrización representa un exceso de liquidez.

Por lo anterior se concluye que no existe exposición significativa al riesgo de liquidez porque ni en la primera ni en la segunda banda de tiempo se registró un IRL acumulado inferior a uno (100%). Es importante resaltar que presentar un exceso de liquidez usualmente, afecta las utilidades de la entidad debido a una disminución en los rendimientos de los activos. Se presenta esta referencia dado que al hacer un estudio de un caso específico se plantea soluciones por medio del desarrollo de un sistema que permite cuantificar un valor del IRL que dado al caso se aporta conocimiento en la arquitectura de microservicios que se trabaja en el desarrollo que se lleva a cabo tanto en el front-end como back-end.

Además, Nebel (2018) en “Arquitectura de Microservicios para plataformas de Integración” aplicado en un Ámbito empresarial y haciendo uso de caracterización de escenarios, Impacto de características sobre PI, impacto de Arq. de microservicios sobre preocupaciones, idoneidad de escenarios para Pis basadas en microservicios. El objetivo general de esta tesis es estudiar la aplicabilidad de una arquitectura de microservicios para la construcción de Plataformas de Integración.

Por lo tanto, se obtiene como resultado un conjunto de características de los escenarios de integración, que permite identificar si un escenario se beneficia al utilizar una PI basada en microservicios, impacto en la calidad de la PI al aplicar una arquitectura de microservicios y se concluye que la arquitectura de microservicios es aplicable para la construcción de PIs en escenarios con determinadas características, el área de microservicios carece aún de madurez y estandarización, Dado que se abordan los conceptos de microservicios aplicados en el ámbito empresarial por medio de estudios y comprobaciones de viabilidad.

De este modo brindando como aportes microservicios, contenedores y virtualización, plataformas de integración, arquitectura y calidad de software, por último, se asemeja en la arquitectura y el estudio de microservicios, dado que se nos presentan escenarios con los cuales se puede tomar decisiones al momento de tomar determinada zona en la cual trabajar.

Por otra parte, Galicia y Ricaurte (2020) en su trabajo “Desarrollo de un sistema de información basado en cloud computing para la gestión de activos de la universidad piloto de Colombia.”que se aplica a la Universidad Piloto de Colombia partiendo de una metodología de Tipo de investigación correlacional: cualitativa y cuantitativa porque mientras se hace uso de herramientas informáticas, estadísticas y matemáticas para obtener resultados también se hace una exploración más alta del panorama en un tipo de investigación cualitativa. Teniendo un objetivo de desarrollar una propuesta de un sistema de información para los activos tecnológicos del área de audiovisuales de la UPC, el cual busca mejorar y automatizar la administración de dichos activos a partir de la transición del modelo actual a la integración de un aplicativo Web y una BDD alojados en la nube (Cloud Computing).

Luego de esto, finalmente se presenta como resultado la interfaz web y el diseño de una interfaz móvil, y se concluye que: la creación de los requerimientos funcionales de un sistema son quizás la parte más importante del inicio de un proyecto, pues de este depende que no haya imprevistos en temas vitales como costos y tiempos de entrega. Puesto que la analítica de datos es un tema que surge en el siglo XXI como base fundamental de la optimización de recursos y procesos, es por eso que, este sistema busca, además de arrojar datos para analizarlos y aprovecharlos, la computación en la nube, que es uno de los avances tecnológicos más sustanciales de todos los tiempos, y se toma la referencia ya que presenta el diseño de una web y una interfaz móvil todo ello desarrollado en angular, y nos brinda aportes tales como cloud computing que es un sistema de computación en la nube que ayuda con los servicios y que la información este respaldada en la nube, que se asemeja en la implementación de una web e interfaz móvil con microservicios y se hace en un entorno educativo como lo es una universidad.

De la misma forma, Osorio, Morales (2017) en su trabajo “Desarrollo de una API para el análisis de inversiones y créditos a través de factores de interés compuesto, implementando microservicios.” Que fue aplicado en un Ámbito empresarial, se tiene como objetivo desarrollar un servicio de análisis financiero para la evaluación de viabilidad económica de proyectos que requieran inversión o crédito implementado en la nube, obteniendo como resultado que se desarrolló una API para la estimación del valor presente neto y del análisis de beneficio/costo de proyectos, y que cuenta con un servicio asíncrono encargado del cálculo de los factores presente dado futuro, presente dado anualidad y presente dado gradiente aritmético, está dividida en microservicios que se ejecutan en ambientes aislados y controlados, y cada uno cumple una tarea específica y fundamental.

Esto permite dar a conocer la API de análisis financiero con instituciones que brindan apoyo a las nuevas, micro y pequeñas empresas, da paso a implementar procesos de entrega continua para facilitar la liberación de nuevas versiones de los microservicios que componen a la API, además a publicar los contenedores de la API en un servicio de infraestructura o plataforma en la nube que soporte una fácil instalación de los mismos, así mismo se toma en cuenta muchos aspectos importantes como lo son conceptos de Ing. económica, interfaces de programación de aplicaciones, arquitectura de microservicios, análisis de viabilidad financiera, y aporta el desarrollo de una API, así como también su documentación y utilización de la misma, con el tema actual se asemeja en cuanto a la utilización de un framework api como RPC implementando los microservicios, del mismo modo aporta también en algo de seguridad informática y en procesos de desarrollo web por medio de contenedores como Docker.

También, López (2020) en su trabajo “Propuesta de arquitectura de microservicios para monitorear la calidad de los procesos de lanzamiento de software corporativo” nos muestra aplicado en el Ámbito empresarial, esta empresa del sector bancario y se parte de la realización de unos requisitos, restricciones y modelo arquitectónico luego de ello se hace la implementación de un prototipo funcional. Ya que, se parte de un objetivo principal que es analizar, diseñar e implementar un sistema con características específicas de flexibilidad, escalabilidad, velocidad de respuesta y facilidad de mantenimiento, que le dé la posibilidad a una gran empresa del sector bancario a tener un producto consolidado para procesar y mostrar datos de diferentes fuentes y relacionarlos.

De tal forma, obteniendo como resultado que Quality Data Meter fue un proyecto que logró la mayoría de sus objetivos a nivel técnico, y concluyendo que las tecnologías, frameworks y herramientas utilizadas fueron de gran apoyo para el desarrollo ydistribución de la aplicación; por ende, se considera que han traído ventajas, y dadas las características encontradas en el producto final, se puede decir que en cuanto a las propiedades del software se han cumplido plenamente los requisitos.Esta referencia se toma en cuenta porque aporta información sobre los microservicios y pagina web, y brinda aportes tales como evidencia información relevante como el uso de un framework para el desarrollo web y el uso de los microservicios dentro de una organización bancaria. Por otra parte, se asemeja en el uso de microservicios para el desarrollo de una página web y existe una diferencia en cuanto a la empresa que se aplica ya que es un sector bancario.

Por otra parte, Escobar (2017) en “Sistema de información web basado en la arquitectura orientada a servicios (SOA) para la gestión de usuarios, pacientes, proveedores, historitas clínicas y reportes de la fundación IPS para la salud física y metal fime.” Realizado en FUNDACIÓN IPS PARA LA SALUD FÍSICA Y MENTAL FIME partiendo de la metodología SOMA, que tiene como objetivo desarrollar un sistema de información web basado en la arquitectura orientada a servicios para la gestión de usuarios, historias clínicas y generación de reportes para la IPS FIME. Se tiene resultados como: Al implementar esta aplicación basada en la arquitectura orientada a servicios se expone a la Fundación IPS FIME a un desarrollo de vanguardia y competitivo a nivel empresarial.

Al dar por finalizado este proyecto, se optimizaron procesos como el registro, actualización, consulta y eliminación de información, con criterios estandarizados de la propia IPS, y se concluye que el desarrollo de aplicaciones orientadas a servicios es un claro ejemplo de los modelos de negocio actuales. La evolución de estos, ha sido tal, que ya no se centran solamente en ofrecer productos si no, en la calidad y mejoramiento de un servicio, de este modo se toma en cuenta ya que se establece un diseño basado en la arquitectura de servicios, brindando aportes como el desarrollo web, y optimización en una fundación IPS por finalizar se asemeja en el mejoramiento de las funciones internas de la organización y se puede tomar como diferencia el hecho de que se trabaje sobre servicios y no microservicios.

Finalmente, Grzegorz , Ojdowska, y Przybilek(2021) en “Arquitectura Monolítica frente a microservicio: una evaluación de rendimiento y escalabilidad.” Aplicado en Empresas interesadas en las nuevas tecnologías de microservicios, se tiene como objetivo comparar el rendimiento y la escalabilidad de arquitecturas monolíticas y de microservicios en una aplicación web de referencia.

Y dado el caso, se presentan resultados como pruebas ejecutadas en un entorno local (no en la nube) permiten establecer una línea base de rendimiento, en la aplicación de microservicio, las solicitudes pasan entre API Gateway y el servicio de backend, lo que impone una sobrecarga de comunicación que deteriora el rendimiento, y se concluye que en una sola máquina, un monolito funciona mejor que su contraparte basada en microservicios, La plataforma Java hace un mejor uso de máquinas poderosas en el caso de servicios de computación intensiva en comparación con .NET, escalar más allá de un cierto número de instancias degrada el rendimiento de la aplicación.

Por ello, este aporte se toma en cuenta porque en la medida del trabajo a tratar, esta referencia permite aclarar de forma experimentada como el uso de microservicios se diferencia entre el uso de monolitos, partiendo también que aporta una experiencia en cuanto a test de rendimiento de las arquitecturas basadas en monolitos, frente a las arquitecturas que usan microservicios, y finalmente se asemeja en el estudio de los microservicios de tal manera que es una guía para su buen uso y mejorar las practicas que se realicen.

# OBJETIVOS

## OBJETIVO GENERAL

Desarrollar un Aplicativo web y una app móvil que sistematice las reglas de negocio de una biblioteca de una institución educativa mediante la arquitectura orientada a los microservicios.

## OBJETIVOS ESPECÍFICOS

* Caracterizar el desarrollo e implementación de la arquitectura orientada a microservicios en aplicaciones web y móviles a través de la revisión sistemática de literatura.
* Aplicar la arquitectura de microservicios para sistematizar las reglas de negocio y obtener un producto mínimo viable para la biblioteca de una institución educativa.
* Evaluar la calidad y el funcionamiento del aplicativo web y app móvil desarrollado a través de un caso de uso.

# MARCO TEORICO

### REGLAS DE NEGOCIOS.

En un ambiente, tanto empresarial como educativo, la mayoría de las veces, las reglas de negocio toman vital importancia, puesto que en la actualidad se ha incrementado de forma notable la demanda de aplicaciones informáticas, y junto con ello el desarrollo, que se verá afectado en gran medida, de acuerdo a las reglas de negocio que se hayan definido.

De acuerdo con esto unas definiciones que se toma en cuenta es “Las reglas de negocio son los requisitos que más tienden a variar en el tiempo y, de hecho, existen negocios en los cuales las reglas pueden ser muy cambiantes. De manera general se puede afirmar que, en enfoques tradicionales de desarrollo de software, las reglas del negocio se implementan dentro de la capa de lógica de negocio, en diferentes puntos del código fuente.” [10]

Así como también, García, nos da a entender que las reglas de negocio son apartados que nos permiten definir las condiciones para establecer un negocio o precisar de qué forma se controlará el comportamiento de los eventos dentro de este [10] dicho de otra manera, una regla de negocio es una declaración que define, establece y obliga cierto aspecto del negocio[10] , con esto ser pretende imponer la estructura del negocio y básicamente de esta forma condicionar el comportamiento de este.

Esto nos permite conocer que mediante las reglas de negocio en una organización se controlara y se definirá los procesos [10], y sobre esto que un evento o negocio va fracasar cuando no se puedan cumplir las reglas que se han establecido previamente, junto con ello los lideres del negocio van a determinar las diferencias que existen entre un evento exitoso o uno infructuoso. [10], tomado de [11] y [12].

Así de este modo una regla de negocio, básicamente, es una declaración compacta sobre un aspecto del negocio, esto usando un lenguaje simple, inequívoco, y este a su vez es accesible a todas las partes que se encuentren interesadas, en dado caso a el dueño del negocio, el analista , el arquitecto técnico, llevado al caso actual tenemos partes interesadas como lo son el rector del colegio, los encargados de la gestión bibliotecaria y finalmente los estudiantes que serán los participes activos en el proceso.[10] tomado de [13].

Ahora bien, las reglas de negocio se identifican durante el transcurso de la captura y análisis de requisitos, básicamente una buena regla de negocio es cohesiva, esto quiere decir que describe solamente un concepto [10],asegurándose de que las reglas de negocio sean cohesivas, se hacen más fáciles de definir y aumenta la probabilidad de que sean reutilizadas [10] tomado de [14].

Según lo antes mencionado las reglas de negocio deben centrarse en un solo concepto, a menudo en una aplicación a desarrollar se identifican gran cantidad de reglas, en dado caso en la biblioteca se toman en cuenta los requerimientos por parte de las personas que administran estos recursos y de los estudiantes que quieren el acceso a la información, por esto es recomendable que las reglas de negocio deban ser separadas de los demás artefactos de la captura de requisitos ya que pueden ser referenciadas desde dentro de los artefactos como los casos de usos o los diagramas de clases.[10]

En ocasiones se definen las reglas dentro de los casos de usos que tendrá la aplicación móvil o web, lo cual se puede convertir en una tarea compleja, además de esto se debe considerar que una misma regla de negocio puede ser referenciada en más de un caso de uso, un ejemplo claro es cuando una regla depende de otra o se relacionan de algún modo. [10] tomado de [11], por lo tanto, las reglas de negocio deben ser atómicas, o sea no pueden ser descompuestas sin que se pierda información, al igual que no pueden ser ambiguas, por el contrario, deben ser compactas, por lo general son frases cortas, además de ello deben ser consistentes y compatibles, estas son algunas características universales y se aplican a cualquier lenguaje o dominio de aplicación,[10] tomado de[14].

El enfoque que tienen las reglas de negocio define un conjunto de pasos y técnicas para desarrollar un sistema basado en las reglas de negocio, que a su vez es un sistema automatizado que separa las reglas de forma lógica, y quizás físicamente, de otros aspectos del sistema que se comparten por medio de datos almacenados, interfaces de usuario y/o aplicaciones, [10] tomado de [15].

Finalmente, en el proceso de desarrollo de aplicativo web o móvil, siguiendo el enfoque basado en las reglas de negocio, este está compuesto por varios pasos que son comunes a varias metodologías existentes, aunque la diferencia radica en cómo se deben tratar las reglas de negocio en cada uno de los pasos, teniendo en cuenta esto los pasos que se tienen son los siguientes:[10] tomado de [15]

#### Definir el Alcance del Negocio:

Se captura los requisitos y se establece los objetivos del negocio, siempre teniendo las reglas de negocio como clave para la definición de los procesos de negocio.

#### Descubrimiento:

Se descubre las decisiones importantes que están detrás de los eventos de negocio.

#### Análisis:

Se realiza un análisis de los datos, reglas y procesos, profundizando más en e análisis de reglas.

#### Diseño:

Se diseña el sistema basado en reglas de negocio, para lo cual se integran tres aspectos fundamentales: los datos, las reglas y los procesos.

#### Implementación y Prueba:

Se relaciona directamente con la herramienta que se selecciona para manejar las reglas de negocio, se propone como la selección de una tecnología o herramienta existente, lo cual lo hace más sencillo de transitar y más ágil.

#### Programación Declarativa:

Las reglas permiten decir qué hacer y no cómo hacerlo

#### Separación de los datos y la lógica:

La lógica del negocio puede ser mucho mas fácil de entender, mantener y enfrentar posibles cambios en el futuro, pues toda la lógica está representada en las reglas.

#### Conocimiento centralizado:

Usando las reglas, se crea un repositorio de conocimiento que permite verificar la política de negocio por todo el personal del negocio.

#### Simplicidad:

El enfoque es simple de entender tanto el personal del negocio como para el público interesado.

#### Facilita el desarrollo de la aplicación:

El experto define las reglas que se ejecutan en aras de satisfacer los requerimientos de la aplicación, sin necesidad de modificar el código fuente un otro aspecto técnico.

#### Agilidad:

Se trata de ofrecer simple respuesta y rápida ante los requisitos dinámicos. [10] tomado de [16]

### MICROSERVICIOS.

#### Definición

Los microservicios se conciben como “un estilo arquitectónico enfocado a desarrollar aplicaciones mediante un conjunto de servicios, independientes, escalables, colaborativos, evolutivos, capaces de auto adaptarse a ecosistemas complejos.” [17], a su vez los microservicios se conciben como un estilo arquitectónico enfocado en desarrollar una aplicación web o móvil, mediante un conjunto de servicios, estos serán independientes, escalables, colaborativos, evolutivos capaces de auto adaptarse a ecosistemas complejos.[17].

Los microservicios surgen con la finalidad de resolver las limitaciones que presentaban los sistemas monolíticos, los cuales se caracterizan por mantener los servicios como una unidad lógica [9], utilizando los mismos recursos computacionales. Un gran porcentaje de sistemas de información en el mundo, hoy en día se encuentran construidos bajo este enfoque monolítico, y en vista de las restricciones que mantiene el evolucionar, conlleva a que las organizaciones apunten a migrar sus ecosistemas a un enfoque ya modernizado, evolutivo y con las infraestructuras en la nube, que a su vez tienen grandes beneficios,[17].

De igual forma Ortiz Noriega, da a conocer que los microservicios que es un patrón que busca construir una aplicación ya sea web o móvil como si se tratara de un conjunto de pequeños servicios, que a su vez son elementos que componen una aplicación y se organizan de acuerdo a funcionalidades específicas,[18] tomado de [19], por lo que en las aplicaciones puedan ser desarrolladas, testeadas y mantenidas con mayor facilidad.

Finalmente, los microservicios se consideran como una arquitectura de software que permite la descomposición de un sistema, sus componentes o sus funcionalidades en un conjunto de pequeños servicios, que se implementan, despliegan y gestionan de forma independiente,[20]

#### Composición

Una arquitectura que se basa en microservicios tiene tres capas: la capa de datos, que tiene toda la información, la capa de microservicios, que se compone por todas las APIs que implementan funcionalidades de la aplicación y los servicios RESTful que ayudan a responder a las peticiones de los clientes; y por último la capa de presentación, que está compuesta por la interfaz o interfaces que facilitan la interacción del usuario con el sistema.[18] tomado de [21]

### LENGUAJES DE PROGRAMACIÓN.

Un lenguaje de programación puede definirse como un conjunto de sintaxis y reglas semánticas que definen a los programas del ordenador. Es la forma estándar de comunicación con la cual podemos entregarle instrucciones al computador. Un lenguaje de programación es la herramienta que le da la capacidad al programador de especificarle al computador, qué tipo de datos actúan y que acciones tomar bajo una variada gama de circunstancias. Un programa escrito en un lenguaje de programación cualquiera, necesita pasar por un proceso de compilación o interpretación, para de esta forma ser traducido al lenguaje de máquina y posteriormente pueda ser ejecutado por el computador. [22]

#### Historia

En los inicios la programación resultaba muy costosa y lenta, debido a que solamente existía un lenguaje de programación, el cual era el lenguaje máquina o código binario, por consiguiente en los años 50 se comenzó a crear una escritura más simbólica, que recibió el nombre de código ensamblador (Assembly), la cual hacía uso de abreviaturas mnemotécnicas para poder facilitar operaciones; a pesar de esto la traducción al lenguaje máquina seguía siendo manual, pero tiempo después apareció el programa ensamblador o Assembler el cual se encargaba de hacer esta traducción ([23],2022 tomado de [24],2004)

Con la expansión del uso de los ordenadores , surgió la necesidad de sustituir a esos lenguajes por otros que facilitaran el trabajo de los desarrolladores, aparecieron entonces los llamados lenguajes de alto nivel, los cuales resultaban ser más fáciles de usar para las personas, los primeros lenguajes de alto nivel aparecieron en la década de los 50, con John Backus, quien desarrolló el programa SPEEDCODING, dando en 1956 de esta forma origen a FORTRAN1, lenguaje que fue pensado para poder resolver problemas técnicos y científicos ([23],2022 tomado de [24],2004).

A finales de los años 50, era difícil portar un programa de un ordenador a otro, leer códigos y modificarlos, por tal motivo desde el Departamento de Defensa de los Estados Unidos, se organizó una conferencia sobre los lenguajes (Conference on Data Systems Languages) de la cual surgió las especificaciones de desarrollo para COBOL2 ([23],2022 tomado de [24],2004)

En el año de 1964, Thomas E. Kurtz y John G. Kemeny profesores del Dartmouth College desarrollaron un lenguaje que ayudará a los estudiantes a poder introducirse en los sistemas de tiempo compartido, lenguaje que fue llamado BASIC debido a lo sencillo que era comparado con lenguajes existentes y su uso se extendió tanto para tareas administrativas como para problemas más científicos ([23],2022 tomado de [24],2004)

En 1969 los investigadores Dennis Ritchie y Kenneth Thompson crean UNIX. Un año más tarde Thompson creó un lenguaje que denominó B, y dos años más tarde su compañero Ritchie basándose en B creó C, él fue el lenguaje más portable del mercado de esa época, posteriormente en los años 80 Bjarne Stroustrup creó una ampliación del lenguaje C, que se acabó convirtiendo en 1984 en el compilador de C++ ([23],2022 tomado de [24],2004)

Niklaus Wirth, profesor del Instituto Politécnico Federal de Zurich creó PASCAL (estandarizado en 1983), el cual permitió que muchas personas se introdujeran al mundo de la programación debido a que este lenguaje facilitaba el aprendizaje utilizando programación estructurada, recursividad, punteros, descomposición modular, etc. ([23],2022 tomado de [24],2004)

En 1990 James Gosling basándose en C y C++ creó JAVA, uno de los lenguajes más utilizados. Este lenguaje quería crear una interfaz más intuitiva para la electrónica de consumo como son las calculadoras, las televisiones y otros dispositivos, Debido a que esta industria no evolucionó, se empezó a utilizar JAVA para Internet y se modificó para este uso ([23],2022 tomado de [24],2004)

De este momento en adelante han empezado a aparecer un gran número de lenguajes de programación entre los cuales están los que actualmente conocemos.

#### Clasificación

No hay una única forma de clasificar a los lenguajes de programación, debido a que existe una gran variedad de criterios por los cuales se los puede organizar. A Continuación, se muestra y se describe algunos de los criterios de clasificación más importantes

##### Según el proceso de traducción

El proceso de traducción es el proceso al que se somete un código fuente escrito en un lenguaje de programación de alto nivel antes de convertirlo en un lenguaje de máquina de bajo nivel. Básicamente hay de 2 tipos, los lenguajes de programación compilados y los interpretados. ([25], 2017)

###### Lenguajes compilados

Para este tipo de lenguajes, es un compilador el que realiza la tarea de traducir el código fuente a código máquina para una plataforma específica. El compilador traduce el código fuente de un programa al lenguaje nativo del ordenador, antes de la ejecución del programa. ([25], 2017 Tomado de [26], 2017)

El paso de compilación produce un archivo intermedio, también llamado archivo de código objeto. Este archivo tiene códigos de instrucción que en esencia representan las funciones de la aplicación. Ejemplos de este tipo de lenguajes son C, C++. ([25], 2017)

###### Lenguajes interpretados

En estos lenguajes la traducción del código fuente no ocurre de antemano, es decir, la traducción se produce simultáneamente con la ejecución del programa. El intérprete comienza a traducir cada instrucción al instante de la ejecución, por lo tanto, el código fuente de un lenguaje interpretado se ejecuta directamente en la plataforma de destino por un programa o intérprete [10]. Como ejemplo de este tipo de lenguajes podemos citar a Python, PHP. [25] , 2017 Tomado de [27],1995)

Según el paradigma

Todos los lenguajes de programación se guían por un enfoque en su utilización que se conoce como paradigma, dicho de otra forma, es una perspectiva desde la cual se mira el problema y se resuelve con posibilidades específicas de las tecnologías y dispositivos modernos de manera que se haga más sencilla, manejable y efectiva dicha solución. [28] , 2020 Tomado de [29], 2018)

###### Paradigma funcional

Una función es la unidad mínima o célula de trabajo de un programa funcional. Las funciones se pueden describir como un fragmento de código con cierto grado de independencia (Felleisen et al., 2005), que cumple un objetivo específico y que puede retornar valores. Uno de los objetivos del paradigma funcional es que la implementación e interacción de diferentes funciones que cumpliendo cada una un pequeño objetivo específico, en conjunto y unidas apropiadamente logren alcanzar la solución que el problema requiere. ([30],2018 tomado de [31], 2005)

La programación funcional puede resumirse en que esta consiste en gran medida en definir funciones que devuelven un valor único, correspondiente a cada conjunto válido de argumentos, y que además una llamada a una función no tiene por qué generar efectos secundarios que puedan alterar otros cálculos. ([32], 2014)

###### Paradigma imperativo

La programación imperativa (y por ende sus lenguajes) se basa en el concepto de estados, cuyo elemento principal es la variable, que básicamente puede definirse como el espacio de memoria en donde se almacenan datos y cuya interacción se logra mediante la ejecución de unas instrucciones de un lenguaje de programación alcanzando determinados valores que al final, son los que resuelven un determinado problema ([33] 2006). La programación imperativa es la más antigua que ha existido en el mundo de la programación y ha sido muy útil para encontrar soluciones a problemas de índole matemático. ([34],2017 Tomado de [33], 2006)

###### Paradigma orientado a objetos

En el paradigma de la programación orientada a objetos, el objetivo pasa de modelar el comportamiento del mundo a modelar los objetos que en él existen junto a sus comportamientos e interacciones. La arquitectura de la aplicación está de esta forma, mucho más cerca de la estructura real del problema. La programación orientada a objetos es una nueva forma de pensar los problemas usando modelos sobre conceptos del mundo real. Lo principal es construir objetos, los cuales combinan estructuras de datos y comportamientos como una única entidad. ([35]., 2018)

En la programación orientada a objetos, todas las entidades con las que computamos son un objeto, objetos que deben estar bien estructurados para adaptarse a los objetivos de nuestros cálculos. La herramienta clave es la definición de clase, la cual indica que los objetos que pertenecen a esta clase comparten estructura definida por las propiedades que todos tienen. Una clase puede también heredar de una superclase más simple, de manera que un objeto de esta clase es también un objeto de la superclase. Finalmente, para calcular con objetos, se pueden definir métodos que sólo se utilizan cuando los objetos son de ciertas clases. ([32], 2014)

Según el nivel

Lenguajes de bajo nivel

Generalmente los utilizan los mismos fabricantes y su comprensión es muy complicada, son llamados también lenguajes ensambladores traduciendo estos a órdenes que son ejecutados por la máquina (lenguaje de máquina). Son realizados bajos esquemas mnemónicos. ([36] , 2019)

###### Lenguajes de alto nivel

Los lenguajes algorítmicos están diseñados para expresar cálculos matemáticos o simbólicos. Pueden expresar operaciones algebraicas en notación similar a las matemáticas y permiten el uso de subprogramas que empaquetan operaciones de uso común para su reutilización. Fueron los primeros idiomas de alto nivel. ([36] , 2019)

#### Áreas de los lenguajes de programación

Hoy en día, la programación (y con ello los lenguajes de programación) se encuentra en todas partes, ante esto se puede diferenciar algunas áreas, ramas o campos en los cuales podemos encontrar los lenguajes de programación. A continuación, se describe algunas de las áreas más significativas

##### Lenguajes para desarrollo Web

Los lenguajes de programación para la web permiten crear aplicaciones en las cuales los usuarios pueden acceder a un servidor Web a través de la red (sea una intranet o internet) mediante un navegador determinado ([37], 2018 Tomado de [38], 2002.)

El desarrollo de aplicaciones web se lo puede separar en dos grandes partes, la parte comprendida por el Front-end y el Back-end

###### Front-End

Los lenguajes de la parte Front-End se encargan del diseño y desarrollo web que funcionan en el navegador, y son en esencia responsables de interactuar con los usuarios; para ello se utilizan tecnologías como HTML5 para estructurar el contenido del documento, CSS para los estilos y JavaScript como lenguaje de programación para darle interactividad a los diseños. ([39], 2021 tomado de [40], 2020)

###### Back-End

Los lenguajes de la parte Back-End se encargan de desarrollar la parte lógica de una aplicación Web, estos lenguajes deben asegurarse de que todo funcione correctamente; son responsables de recibir los datos, procesarlos y de enviarlos al cliente, en esencia se encargan de las consultas a la base de datos y del contacto con el servidor ([39], 2021 tomado de [41], 2019).

Lenguajes para desarrollo Móvil

Según Cárdenas Villavicencio et al. ([42],2021) “Las aplicaciones móviles son piezas de software destinadas al despliegue y funcionamiento en sistemas operativos de dispositivos móviles, ya sean estos IOS o Android”.

Básicamente con los lenguajes de programación para desarrollo móvil podemos hacer tres tipos de aplicaciones móviles: aplicaciones nativas, aplicaciones web y aplicaciones híbridas; las aplicaciones nativas se desarrollan para un sistema operativo, mientras que las aplicaciones web son sitios web que se parecen a las aplicaciones nativas, y por su parte las aplicaciones híbridas combinan aspecto de las anteriores y funcionan tanto en un entorno web como un sistema operativo móvil( [42] , 2021 tomado de [43] , 2015)

##### Lenguajes para Machine Learning

La programación del aprendizaje automático o Machine learning es una parte de la inteligencia artificial que no se suele asociar con la programación como tal, debido a que en realidad se trata de un conjunto de algoritmos que se usan en las aplicaciones, sumado a que esta disciplina se ocupa del aprendizaje automático de grandes volúmenes de datos. ([23],2021)

En la actualidad se pueden encontrar varios lenguajes de programación que se utilizan para el desarrollo de aplicaciones de inteligencia artificial y machine learning; es común que las aplicaciones de este tipo tengan requisitos muy específicos que solo algunos lenguajes de programación puedan solventar, debido a esto los lenguajes de programación de machine learning han tenido que evolucionar, y otros se han creado en función de los requisitos únicos de las aplicaciones de IA; como ejemplos de estos lenguajes de programación podemos encontrar a Python, R, Julia y Matlab ([44], 2020)

Lenguajes para Videojuegos

Para el desarrollo de videojuegos, los lenguajes de programación son un elemento esencial, debido a que hace más sencilla la creación de escenarios y la comunicación del usuario con la maquina; un videojuego es en esencia un tipo de software que tiene ciertas restricciones, ciclos y decisiones tomadas que están en función con los datos ingresados o circunstancias que se presentan dentro del juego durante la interacción del usuario con la computadora o consola. Entre los diversos lenguajes para programar videojuegos los más conocidos son c, c++, c#. etc. ([45], 2018)

##### Lenguajes para Escritorio

Las aplicaciones de escritorio son programas que se instalan directamente en el ordenador con la principal característica de que hacen uso completo de los recursos de la computadora cliente en la que se ejecutan; a diferencia de las aplicaciones web este tipo de aplicaciones no necesitan conexión a Internet o a cualquier tipo de red para poder funcionar. ([46], 2022)

##### Lenguajes para sistemas Embebidos

Los sistemas embebidos no son equivalentes a los sistemas de cómputo usados en computadoras de escritorio que comúnmente conocemos(principalmente porque estos sistemas cuentan con recursos limitados); el proceso de programación de un sistema embebido requiere inicialmente un IDE en donde se realizar la programación de las instrucciones; cundo el programa es finalizado se hace uso de un compilador, el cual transforma el texto que contiene las instrucciones en código de máquina; generalmente el programa que ejecuta un sistema embebido es elaborado en lenguajes como el ensamblador, ANSI C, C++ o Basic ([47], 2015).

### BIBLIOTECAS.

Según Gallo León [48], las bibliotecas aparecieron para asegurar el acceso a los libros, independientemente de su formato (códice, rollo, tableta…). La preservación sólo era una parte de esa misión. Por tanto, su función primera fue la de servicio de información, de herramienta de acceso a la información para ayudar al conocimiento.

Las bibliotecas, son la memoria colectiva de la raza humana, en los libros de las bibliotecas esta todo aquello que nuestros predecesores consideraron que era lo bastante importante como para ser escrito y preservado; nuestros logros culturales humanos se registran principalmente por escrito y mediante gráficos preservados para las generaciones venideras, de esta forma las bibliotecas nos han permitido comprender nuestros monumentos y artefactos e interpretar su contenido y el contexto en el cual existieron ([49], 2014).

**IX. RESULTADOS**

1. **CARACTERIZACIÓN**

KOHA es un Sistema Integrado de Gestión de Bibliotecas (LIMS) de código abierto muy utilizado hoy en día por cientos de bibliotecas públicas, escolares y académicas de todo el mundo, fue creada para poder realizar un seguimiento de las operaciones de la biblioteca, tales como artículos, facturación, seguimiento de diversos artículos propiedad de la biblioteca, tales como libros, revistas, periódicos, etc; puede también proporcionar diversos servicios bibliotecarios a los usuarios, como la circulación de material bibliotecario, la catalogación, la llegada de nuevos usuarios, etc.[50]

El desarrollo de KOHA comenzó en 1999, financiado por un grupo de bibliotecas de la zona rural de Nueva Zelanda; fue diseñado para funcionar con un mínimo de recursos de hardware y se ejecuta en el sistema operativo Linux junto con el servidor web apache; utiliza el popular sistema de gestión de bases de datos de código abierto MySql y está escrito en Pearl.[50]

VuFind es un desarrollo de la Universidad de Villanova de Pensilvania, EE.UU; el proyecto desde sus comienzos en 2007 contó con la participación de bibliotecarios e informáticos en su desarrollo; VuFind permite hacer búsqueda de resultados detallados, lo que le permite al usuario refinar los resultados de búsqueda mediante filtros permitiendo explorar los resultados en muchas formas, por institución, clasificación, idioma, materias, formato y autor; aporta ademas, sugerencias de recursos de acuerdo a los registros visualizados.[51]

Es compatible con los software de gestión de referencias, como por ejemplo Zotero, para que el usuario pueda guardar los resultados de búsqueda y etiquetarlos en el mismo y se conecta con el SIGB, como por ejemplo Koha, para que el usuario pueda encontrar el resultado del documento con el estado y la ubicación en vivo.[51]

Para su correcto funcionamiento requiere contar con: Apache 2.2.12+, PHP 5.6+, MySQL 5.5+ y Java JDK 1.8+, y cuenta también con una Interfaz simple para dispositivos móviles que facilita el acceso al recurso desde Smartphones[51]

NewGenLib tubo su origen en la India en 2003, como un proyecto de colaboración entre la organización profesional sin ánimo de lucro Kesavan Institute of Information and Knowledge Management37 y una empresa de desarrollo de software comercial llamada Verus Solutions38; este sistema era en un principio software propietario pero se distribuye bajo licencia GPL desde 2008 y es muy utilizado en países en vías de desarrollo.[52]

NewGenLib utiliza la arquitectura cliente-servidor, y utiliza una aplicación cliente escrita en Java, lo cual la hace multiplataforma. En cuanto al servidor, requiere Linux como sistema operativo y PostgreSQL como base de datos. La versión más reciente es la 3.0.3 Update 5, publicada el 23 de abril de 2012.[52]

Las bibliotecas, por otra parte, son el principal apoyo académico en el suministro de información y conocimiento, insumos básicos y necesarios para el proceso de aprendizaje, que realizan tanto el académico o profesor y el estudiante en su papel de aprendiz. Ademas de su función principal de ser proveedora de información, indispensable para la educación, las bibliotecas son también el espacio para el saber, y área privilegiada para el auto estudio[53]

Entre las principales funciones que tienen las bibliotecas se puede mencionar que son un servicio que proporcionan documentos e información en general, brindan también, asesoramiento y consultoría sobre información, y buscan ofrecer en un espacio de trabajo, estudio y convivencia[54]

1. **CONSTRUCCIÓN**

El desarrollo del software se realizó bajo Scrum, metodología que se define como un proceso de administración y control para el desarrollo de software, en el cual se busca dejar de lado la complejidad de las metodologías de desarrollo tradicional, para enfocarse en suplir las necesidades planteadas por el usuario, y mantener a los actores involucrados en el proceso de desarrollo.[55]

El proceso comienza preparando una lista priorizada de items necesarios para lograr el objetivo del proyecto, cada item se conoce generalmente como una historia, que presenta una descripción de la funcionalidad del producto[56]

Scrum estructura el desarrollo en ciclos o iteraciones llamadas Sprints, al comienzo de cada sprint, el equipo selecciona los requisitos del cliente de una lista priorizada llamada product backlog, los sprints tienen una duración fija y nunca se extienden, cada sprint incluye el proceso de planificación, desarrollo y control[56]

Para el desarrollo de este proyecto se utilizaron algunos de los componentes y elementos mas importantes de Scrum, como lo son: roles, historias de usuario, product backlog y sprint backlog. Los resultados de esta implementación se describen a continuación.

1. ***Roles***

Scrum emplea el término de roles para hacer referencia a los papeles que desempeñan cada una de las personas que están involucradas en el desarrollo del proyecto[57]. Para el caso de este proyecto los roles se establecieron en la tabla 1

Tabla 1. Definición de roles

|  |  |  |  |
| --- | --- | --- | --- |
| # | PERSONA | CONTACTO | ROL |
| 1 | Brayan Daniel Ceron | brayan0np@gmail.com | Analista Desarrollador |
| 2 | Daniel Esteban Madroñero | danielmadro@gmail.com | Analista Desarrollador |
| 3 | Deiwer Hernan Chaleal | chalealdeiwer@gmail.com | DBA |
| 4 | Steven Andres Guerrero | stivenguerrero@gmail.com | Scrum Master |
| 5 | JairoAndres Cuchala | cuchalajairo@gmail.com | Product Owner |

***b) Historias de usuario***

Para obtener los requerimientos del software en Scrum se utilizan las historias de usuario, estas son el resultado de la colaboración entre el cliente y el equipo, y básicamente consisten en una breve descripción de las funcionalidades que el cliente necesita que tenga el software[58]. Para el contexto de este proyecto se recopilaron los requerimientos que se encuentran desde la tabla 2 hasta la tabla 11 donde están las historias de usuario para registrar libros, estudiantes y profesores, y las acciones de prestar y devolver libros junto con la gestión de sanciones y generación de reportes:

Tabla 2. Historia de usuario registrar libro

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU01 |
| Nombre | Registrar un nuevo libro |
| Descripción | Como Bibliotecario.  Quiero registrar un nuevo libro  Para llevar registro de todos los libros disponibles |
| Validación | * Todo libro debe tener un titulo, autor, editorial y genero * El sistema debe generar un identificador único para cada uno de los libros |

Tabla 3. Historia de usuario registrar estudiante

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU02 |
| Nombre | Registrar un estudiante |
| Descripción | Como Bibliotecario.  Quiero registrar un estudiante  Para llevar registro de todos los estudiantes que llevan libros |
| Validación | * Todo estudiante debe tener un correo electrónico, dirección y numero de celular para contactarlo * El sistema debe generar un identificador único para cada uno de los estudiantes |

Tabla 4. Historia de usuario registrar profesor

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU03 |
| Nombre | Registrar un profesor |
| Descripción | Como Bibliotecario.  Quiero registrar un profesor  Para llevar registro de todos los profesores que llevan libros |
| Validación | * Todo profesor debe tener un correo electrónico, dirección y numero de celular para contactarlo * El sistema debe generar un identificador único para cada uno de los profesores |

Tabla 5. Historia de usuario registrar préstamo de libro

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU04 |
| Nombre | Registrar préstamo de un libro |
| Descripción | Como Bibliotecario.  Quiero registrar el préstamo de un libro.  Para gestionar el préstamo de los libros |
| Validación | * El préstamo de un libro se debe registrar con los identificadores únicos del estudiante/profesor y del libro * El préstamo de un libro no debe superar las dos semanas * No se debe prestar libros a personas con multas * No se debe prestar mas de dos libros por estudiante/profesor |

Tabla 6. Historia de usuario registrar devolución de libro

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU05 |
| Nombre | Registrar devolución de un libro |
| Descripción | Como Bibliotecario.  Quiero registrar la devolución de un libro.  Para gestionar la devolución de los libros |
| Validación | * La devolución de un libro se debe registrar con los identificadores únicos del estudiante/profesor y del libro * Se debe aplicar una multa,si el libro no se devuelve en dos semanas |

Tabla 7. Historia de usuario gestión de disponibilidad de libros

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU06 |
| Nombre | Administrar disponibilidad de libros |
| Descripción | Como Bibliotecario.  Quiero poder llevar registro sobre todos los libros disponibles  Para poder tener un inventario total de los libros |
| Validación | * Los libros que están prestados no pueden estar en la lista de libros disponibles |

Tabla 8. Historia de usuario eliminar libro

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU07 |
| Nombre | Eliminar libro |
| Descripción | Como Bibliotecario.  Quiero poder quitar de la lista de disponibles a un libro viejo  Para poder dar de baja a los libros deteriorados |
| Validación | * Los libros que se eliminan no pueden estar en la lista de libros disponibles |

Tabla 9. Historia de usuario gestión de sanciones

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU08 |
| Nombre | Gestionar sanciones |
| Descripción | Como Director de la institución.  Quiero que aplicar multas a estudiantes que no hayan devuelto un libro en un periodo de una semana o si lo devolvieron dañado  Para evitar la perdida y deterioro de los libros |
| Validación | * Los estudiantes/profesores con multas no pueden acceder a pedir nuevos libros * Se debe aplicar una multa al estudiante/profesor que no hizo la devolución de un libro a las dos semanas del inicio del préstamo |

Tabla 10. Historia de usuario generar estadísticas

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU09 |
| Nombre | Generar estadísticas |
| Descripción | Como Director de la institución.  Quiero poder llevar registro de los estudiantes y profesores que piden libros en la biblioteca  Para poder tener registro de los estudiantes y profesores que usan los libros de la biblioteca |
| Validación | * Los gráficos deben presentarse como imágenes de buena calidad en pdf * Los reportes no deben contener información sensible de la base de datos |

Tabla 11. Historia de usuario gui intuitiva

|  |  |
| --- | --- |
| Historia de Usuario | |
| ID | HU10 |
| Nombre | Diseñar GUI intuitiva |
| Descripción | Como Director de la institución.  Quiero que la GUI de la aplicación sea amigable e intuitiva  Para que los usuarios puedan usar fácilmente el software |
| Validación | * La combinación de colores de la paleta de colores debe tener alguna relación de armonía y no ser aleatoria * El tamaño y el color de la tipografía debe lucir bien con el resto de la interfaz |

1. ***Product Backlog***

El Product Backlog consiste en un lista dinámica que contiene a todos los requerimientos que se encontraron previamente usando las historias de usuario, en esta lista se le establece a cada uno de los requerimientos la prioridad, la estimación en tiempo y el estado en el cual se encuentra[57].

Para este proyecto, el Product Backlog quedo definido en la tabla 12:

Tabla 12. Product Backlog

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PRODUCT BACKLOG | | | | | |
| ID | H.U | PRIORIDAD | ESTIMACIÓN | SPRINT | ESTADO |
| HU01 | Registrar un nuevo libro | ALTA | 6 | 1 | PENDIENTE |
| HU02 | Registrar un estudiante | ALTA | 6 | 1 | PENDIENTE |
| HU03 | Registrar un profesor | ALTA | 6 | 1 | PENDIENTE |
| HU04 | Registrar préstamo de un libro | ALTA | 5 | 2 | PENDIENTE |
| HU05 | Registrar devolución de un libro | ALTA | 2 | 2 | PENDIENTE |
| HU06 | Administrar disponibilidad de libros | ALTA | 8 | 3 | PENDIENTE |
| HU07 | Eliminar libro | MEDIA | 5 | 3 | PENDIENTE |
| HU08 | Gestionar sanciones | MEDIA | 7 | 4 | PENDIENTE |
| HU09 | Generar estadísticas | BAJA | 6 | 4 | PENDIENTE |
| HU10 | Diseñar GUI intuitiva | BAJA | 5 | 4 | PENDIENTE |

***d) Sprint Backlog***

El Sprint Backlog es también una lista, donde se definen y se planifican las tareas de cada una de las historias de usuarios en Sprints; mediante los Sprints se descompone el proyecto en unidades mas pequeñas, para finalmente establecer el tiempo de terminación , la persona responsable y el estado en el que se encuentra cada una de las tareas[58].

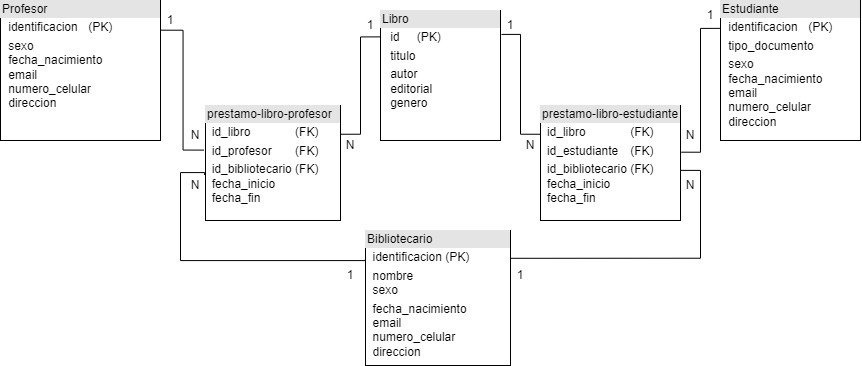
Para este proyecto se establecieron tres Sprints, en el primero de ellos, descrito en la tabla 13, se abarca las historia de usuario de registrar libro, registrar estudiante y registrar profesor:

Tabla 13. Sprint número 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SPRINT BACKLOG | | | | |
| SPRINT # | 1 | | | |
| OBJETIVO | TAREA | RESPONSABLE | ESTIMACIÓN | ESTADO |
| Registrar un nuevo libro | Crear la base de datos | Deiwer Chaleal | 1 | PENDIENTE |
| Crear la tabla “libro” | Deiwer Chaleal | 1 | PENDIENTE |
| Crear la conexión a la base de datos | Deiwer Chaleal | 1 | PENDIENTE |
| Desarrollar el formulario con las entradas necesarias para registrar un libro | Daniel Madroñero | 2 | PENDIENTE |
| Insertar el registro en la base de datos | Deiwer Chaleal | 1 | PENDIENTE |
| Registrar un estudiante | Crear la tabla “estudiante” | Deiwer Chaleal | 2 | PENDIENTE |
| Desarrollar el formulario con las entradas necesarias para registrar un estudiante | Daniel Madroñero | 2 | PENDIENTE |
| Insertar el registro en la base de datos | Deiwer Chaleal | 2 | PENDIENTE |
| Registrar un profesor | Crear la tabla “profesor” | Deiwer Chaleal | 2 | PENDIENTE |
| Desarrollar el formulario con las entradas necesarias para registrar un profesor | Brayan Ceron | 2 | PENDIENTE |
| Insertar el registro en la base de datos | Deiwer Chaleal | 2 | PENDIENTE |
|  |  |  |  |  |

Con base en los requerimientos anteriormente descritos en el Sprint 1, se diseño el modelo entidad relación que mejor representar la intercalación de entidades que participan en el proceso de negocio dentro de una biblioteca. Dicho diagrama se ilustra en la figura 1:

Figura 1. Diagrama entidad relación



El segundo Sprint descrito en la tabla 14, comprende las historia de usuario de registrar préstamo de un libro y registrar devolución de un libro

Tabla 14. Sprint número 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SPRINT BACKLOG | | | | |
| SPRINT # | 2 | | | |
| OBJETIVO | TAREA | RESPONSABLE | ESTIMACIÓN | ESTADO |
| Registrar préstamo de un libro | Crear las tablas “prestamo-libro-estudiante” y “prestamo-libro-profesor” | Deiwer Chaleal | 1 | PENDIENTE |
| Desarrollar el formulario con las entradas necesarias para registrar el préstamo de un libro | Brayan Ceron | 2 | PENDIENTE |
| Insertar el registro en la base de datos | Deiwer Chaleal | 1 | PENDIENTE |
| Generar comprobante del préstamo del libro | Brayan Ceron | 1 | PENDIENTE |
| Registrar devolución de un libro | Desarrollar el formulario con las entradas necesarias para registrar la devolución de un libro | Daniel Madroñero | 1.5 | PENDIENTE |
| Insertar el registro en la base de datos | Deiwer Chaleal | 0.5 | PENDIENTE |

Las historias de usuario para administrar la disponibilidad de libros, y la eliminación de estos, se encuentran en el tercer sprint, descrito en la tabla 15

Tabla 15. Sprint número 3:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SPRINT BACKLOG | | | | |
| SPRINT # | 3 | | | |
| OBJETIVO | TAREA | RESPONSABLE | ESTIMACIÓN | ESTADO |
| Administrar disponibilidad de libros | Desarrollar el dashboard que va a contener la información de los libros disponibles en el momento | Brayan Ceron | 6 | PENDIENTE |
| Crear la consulta que obtendrá la información en la base de datos | Deiwer Chaleal | 2 | PENDIENTE |
| Eliminar libro | Desarrollar el formulario con las entradas necesarias para eliminar un libro | Daniel Madroñero | 1.5 | PENDIENTE |
| Crear la consulta que eliminara un libro en la base de datos | Deiwer Chaleal | 3 | PENDIENTE |
| Mostrar una ventana de dialogo pidiendo la confirmación de la eliminación del registro | Daniel Madroñero | 1 | PENDIENTE |
| Mostrar una ventana de dialogo, donde se notifique la correcta eliminación del registro | Daniel Madroñero | 1 | PENDIENTE |

El cuarto y ultimo Sprint ilustrado en la tabla 16, comprende las historia de usuario para gestionar sanciones, gestionar estadísticas y diseño de interfaz gráfica intuitiva

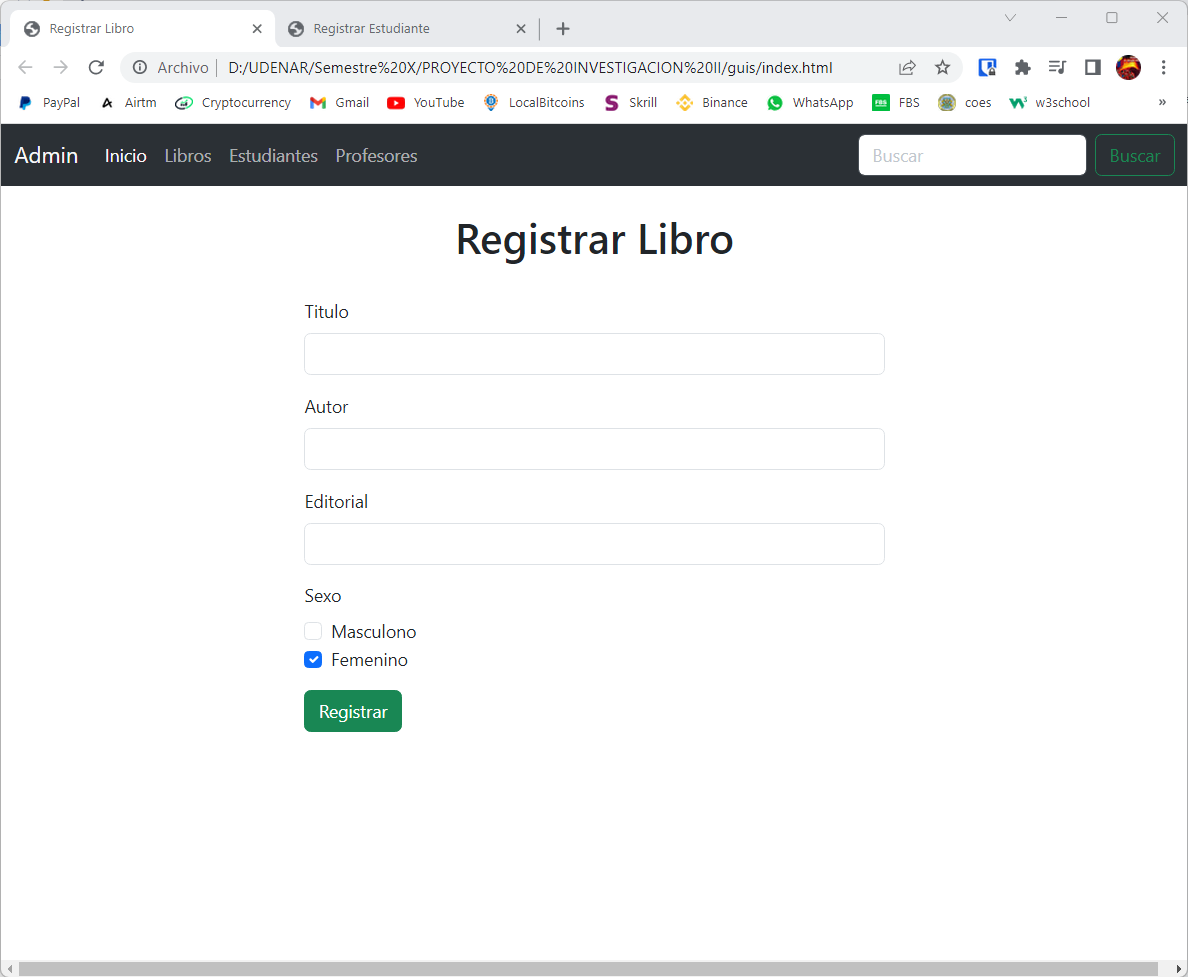
Tabla 16. Sprint número 4:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SPRINT BACKLOG | | | | |
| SPRINT # | 4 | | | |
| OBJETIVO | TAREA | RESPONSABLE | ESTIMACIÓN | ESTADO |
| Gestionar sanciones | Desarrollar la pagina/vista que va a contener la toda información relacionada con las personas que tienen sanciones | Brayan Ceron | 3 | PENDIENTE |
| Crear la consulta que obtendrá la información necesaria en la base de datos | Deiwer Chaleal | 1 | PENDIENTE |
| Generar reporte en pdf con todas las personas sancionada | Brayan Ceron | 3 | PENDIENTE |
| Generar estadísticas | Desarrollar el dashboard que va a contener todos los gráficos de barras, lineas, circulares, etc. | Brayan Ceron | 3 | PENDIENTE |
| Crear las consulta que obtendrán la información resumida en la base de datos | Deiwer Chaleal | 1 | PENDIENTE |
| Generar reporte en pdf con todos los gráficos estadísticos | Brayan Ceron | 2 | PENDIENTE |
| Diseñar GUI intuitiva | Revisar que la paleta de colores, tipografías y el diseño de los componentes en general sea visualmente estético y simple | Daniel Madroñero | 3 | PENDIENTE |
| Comprobar que las configuraciones de las media cueries se vea bien en todas las resoluciones y dispositivos | Daniel Madroñero | 2 | PENDIENTE |

***e) Interfaces gráficas***

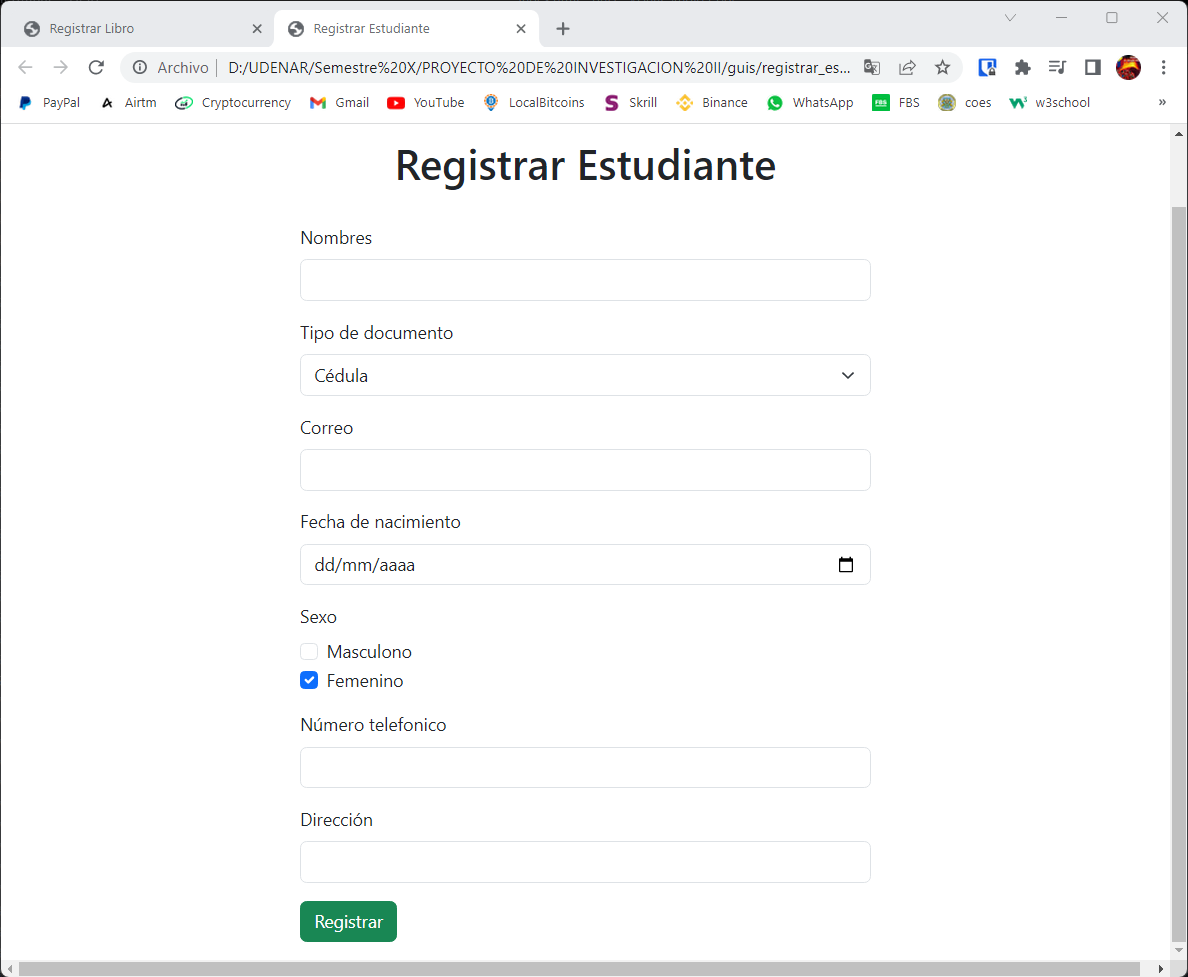
El diseño de las interfaces gráficas de usuario correspondiente al software y a cada una de las funcionalidades descritas anteriormente se ilustran desde la figura 1 hasta la figura 3.Las figura 1 describe la funcionalidad correspondiente al registro de nuevos libros

Figura 1. Interfaz gráfica Registrar Libro



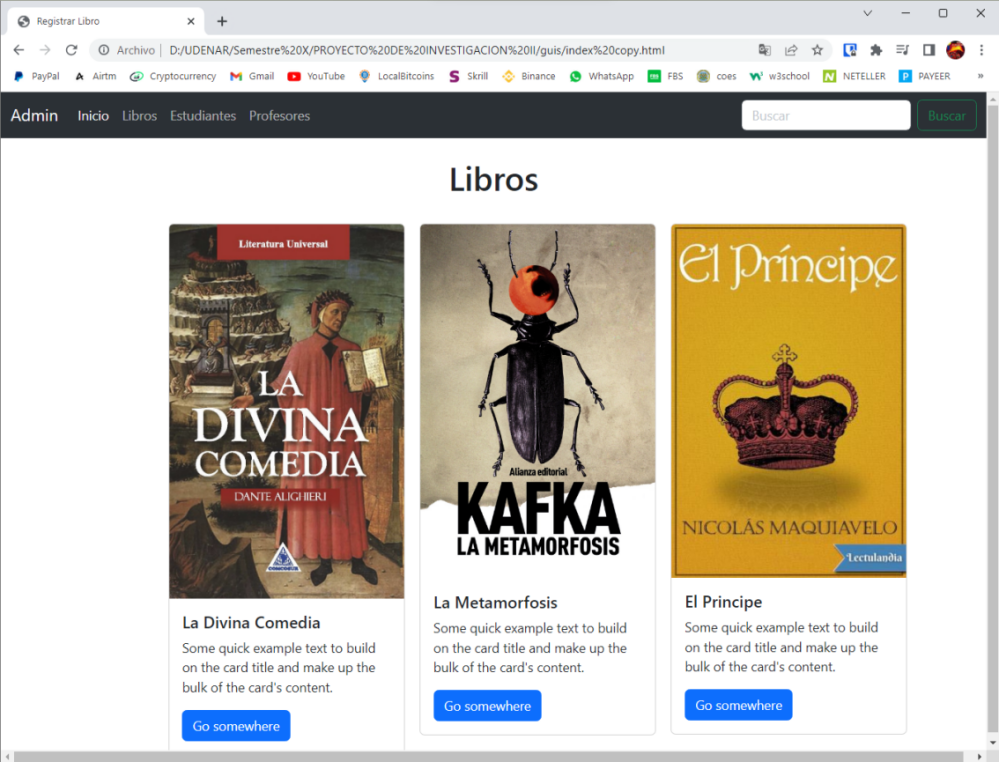
La figura 2 ilustra la interfaz gráfica construida para los requerimientos de registrar un estudiante

Figura 2. Interfaz gráfica Registrar Estudiante



La interfaz gráfica observada en la figura 3 permite evidenciar la disponibilidad de los libros disponibles en el momento

Figura 3. Interfaz gráfica catalogo de libros



1. **VALIDACIÓN**

La implementación y validación del software se realizo en la biblioteca escolar del colegio Liceo De La Universidad De Nariño en la ciudad de pasto, llevándose a cabo durante el ultimo trimestre del año 2022. En este periodo de tiempo, se sometió el software a ejecución en un entorno real y a condiciones de concurrencia y carga de trabajo extremas.

El despliegue del software se llevo a cabo sobre una infraestructura de computo en nube, dicha infraestructura tecnológica era relativamente limitada y pequeña, debido a que el software no requería en un principio gran cantidad de recursos de software, hardware y red.

Se verificó el correcto funcionamiento de cada uno de los módulos del software y se comprobó que este cumpliera con todos y cada uno de los requerimientos pactados inicialmente. En ese orden de ideas, se pudo comprobar y corroborar que el software llevaba a cabo de manera correcta y eficiente las tareas tipo CRUD de creación, obtención, actualización y eliminación de estudiantes, profesores y libros.

De igual forma, para las tareas mas complejas de suministración y devolución de libros, gestión de multas y generación de estadísticas, el software respondió de manera satisfactoria, sin presentar errores, y generando los resultados que se esperaba para dichas operaciones.

Finalmente, durante las primeras dos semanas de implementación del software, se llevo a cabo el proceso de acompañamiento a los usuarios del software (en este caso los bibliotecarios encargados), brindando todo el soporte y asesoría necesaria para que aprendieran a usar y utilizar de forma correcta el software

**XVI. ANÁLISIS DE RESULTADOS**

Desde la puesta en funcionamiento del software, el 5 de octubre del año 2022, hasta el 14 de diciembre del mismo año, el sistema pudo registrar el préstamo y la devolución de al menos 382 libros. Inicialmente solo habían registrados 5 personas, que correspondían en principio a los bibliotecarios, quienes eran los encargados de manipular el software, posteriormente con el registro de prestamos y devoluciones de libros el numero de usuarios creció progresivamente, y se pudieron obtener algunos datos interesantes, que se describen a continuación.

Se pudo observar que los estudiantes que mas libros adquirieron en la biblioteca eran de los grados superiores, principalmente de noveno, décimo y once, los cuales sumaban el 74% del total de libros prestados en el ultimo trimestre del año escolar. De ese 74%, un 25% fue ocupado por el grado once, 28% por el grado décimo y un 21% por el grado noveno; el 26% restante se distribuyó en un 8% para el grado octavo, 7% para séptimo, 4 % para sexto y el 7% restante fue adquirido por los docentes.

El 78% de todos los prestamos se realizan durante el periodo de recreo, tanto en la jornada de la mañana como en la de la tarde, esto debido a que el recreo es el espacio de descanso y esparcimiento que tienen los estudiantes para hacer lo que les gusta; curiosamente los días que mas libros se adquieren es el lunes y el martes respectivamente, esto puede deberse a que los estudiantes buscan aprovechar toda la semana para leer dichos libros.

Como era de esperarse el volumen de prestamos baja considerablemente en los fines de semana a pesar de que la biblioteca este abierta el día sábado, esto sucede porque evidentemente los estudiantes no tienen clases en estos días y aprovechan para descansar.

Las categorías de libros para las cuales se registraron mayor numero de prestamos fueron literatura y biología con un 46% y 35% respectivamente del total de prestamos registrados. Para los libros de literatura, se pudo constatar que los estudiantes se interesan por una gran variedad de obras y novelas de distintos géneros literarios que van desde escritos de acción, aventura , fantasía, romanticismo y llegan incluso hasta temas de índole político y social. Por otra parte, para el caso de los libros relacionados con la biología, se evidencio que los estudiantes buscaban temas relacionados con la ecología, anatomía, medio ambiente y principalmente con la paleontología.

De igual forma, se comprobó que las áreas de menor interés para los estudiantes, son los relacionados con los temas de religión, obteniendo un nulo número de solicitudes por libros de esta clase de temas, sumado ademas, a que la biblioteca cuenta con recursos limitados en su catalogo de libros disponibles para estas temáticas.

Se obtuvo también información valiosa acerca de las categorías de libros que mas multas recibieron, encontrándose que el 80% de los estudiantes que llevaban libros de temas relacionados con matemáticas y química siempre devolvían dichos libros después de la fecha establecida para la entrega. Esto podría explicarse quizás por la dificultad y la complejidad que manejan los temas de estas ciencias.

Como dato curioso, se encontró que había una correlación muy marcada entre la cantidad de libros que un estudiante pedía al mes, y su desempeño escolar, pudiéndose evidenciar que entre mas libros un estudiante adquiriera en la biblioteca, mayor seria su nota y su desempeño en todas las materias en general.

Otro curioso patrón que se puedo obtener, fue que los estudiantes que mas llamados de atención tenían por mal comportamiento, solamente adquirieron libros relacionados con las ciencias sociales y afines

Finalmente, los bibliotecarios también reportaron que no existe la suficiente cantidad de libros para satisfacer la demanda en ciertas áreas y materias, esto se pudo constatar gracias a los reportes que el sistema puede generar, donde en efecto se pudo corroborar que algunos libros nunca llegaban a estar disponibles, debido a la urgencia y premura con la que los estudiantes necesitaban de algunos de estos libros

# Bibliografía

[1] Congreso de Colombia, “Código de la infancia y la adolescencia”, pp. 17–17, nov. 2006.

[2] M. F. Caro Piñeres, R. E. Toscazo Miranda, M. F. Hernández Rozo, y M. E. David Lobo, “Diseño de Software educativo basado en competencias”, vol. 19, núm. 1, p. 98, jun. 2019.

[3] M. Mahrach Mahrach, “Relevancia de las arquitecturas software en la modernización de sistemas heredados.”, 2019.

[4] J. M. Verner y W. M. Evanco, “In-house software development: what project management practices lead to success?”, *IEEE Softw*, vol. 22, núm. 1, pp. 86–93, ene. 2018, doi: 10.1109/MS.2005.12.

[5] J. E. Román Moltó, “Software portable, escalable y extensible para la resolución de problemas de valores propios dispersos de gran dimensión”, 2018.

[6] B. Reynoso, “Introducción a la Arquitectura de Software.”, oct. 2019.

[7] D. López y E. Maya, “Arquitectura de Software basada en Microservicios para Desarrollo de Aplicaciones Web”, jul. 2017.

[8] W. Hasperué *et al.*, “Procesamiento inteligente de grandes volúmenes de información y de flujos de datos”, pp. 229–234, abr. 2021.

[9] C. Saransig y Alexis Fernando, “Análisis de rendimiento entre una arquitectura monolítica y una arquitectura de microservicios – tecnología basada en contenedores”, dic. 2018.

[10] Garcia Juan y André Margarita, “Implementación del enfoque de reglas de negocio utilizando motores de reglas en el desarrollo de  aplicaciones Java”, *Ciencias de la Información*, vol. 46, pp. 41–46, abr. 2018.

[11] S. Ambler, “The Object Primer Agile Model Driven Development with UML 2”, *Cambridge, Cambridge University Press.*, 2004.

[12] O. Rivillas, “ Estado del arte de los motores de reglas de negocio BRM. InéditoTesis de Diploma”, *Universidad de San Buenaventura. Santiago de Cali. Colombia.*, 2012.

[13] B. Lowenthal, “Rule Enabling Applications with Oracle Business Rules”, *Oracle Publication.*, 2005.

[14] R. Ross, “Business Rule Concepts - Getting to the Point of Knowledge. ”, *Business Rule Solutions Inc.*, 2005.

[15] B. von Valle, “Business Rules Applied. Building Better Systems Using the Business Rules Approach.”, 2002.

[16] T. Morgan, “Business Rules and Information Systems:Aligning IT with Business Goals”, *Addison Wesley.*, 2002.

[17] Z. E. Mamani Rodríguez, L. del Pino Rodríguez, y J. C. Gonzales Suarez, “Arquitectura basada en Microservicios y DevOps para una ingeniería de software continua”, *Industrial Data*, vol. 23, núm. 2, pp. 141–149, dic. 2020, doi: 10.15381/idata.v23i2.17278.

[18] K. E. Ortiz-Noriega, J. E. Guevara-Segura, J. P. Santos-Fernández, O. R. Alcántara-Moreno, J. L. Tenorio-Cabrera, y R. J. Sánchez-Ticona, “Aplicación Web con Arquitectura de Microservicios y el Incremento de la Eficiencia en el Comercio Electrónico de una Empresa Peruana de Calzado”, jul. 2022, pp. 158–163. doi: 10.54808/CISCI2022.01.158.

[19] M. H Vural y S. G. Koyuncu, “Systematic Literature Review on Microservices”, *Springer International Publishing,* , pp. 203–217, 2017.

[20] D. Guaman, Lady Yaguachi, C. C. Samanta, J. H. Danilo, y F. Soto, “Performance evaluation in the migration process from a monolithic application to microservices”, en *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, jun. 2018, pp. 1–8. doi: 10.23919/CISTI.2018.8399148.

[21] Z. Mamani Rodríguez, L. del Pino Rodríguez, y J. C. Gonzalo Suarez, “Arquitectura basada en Microservicios y DevOps para una ingeniería de software continua”, *Industrial Data,* vol. 23, núm. 2, pp. 141–149, 2020.

[22] R. Arenaza y E. Roque, “Lenguajes de programación Javascript Java y Javascript. Características. Norma de escritura. Variables y operadores lógicos. Mensajes. Ejercicios. Estructuras condicionales. Funciones y objetos. Aplicaciones”, 2019.

[23] D. de la Fuente, “Un estudio comparativo de los lenguajes de programación de la última década”, sep. 2022.

[24] V. Aranda Trigo, “Historia y evolución de los lenguajes de programación”, vol. 34, pp. 85–95, 2004.

[25] A. Ernest, E. Mensah, y A. Gilbert, “Qualitative Assessment of Compiled, Interpreted and Hybrid Programming Languages”, *Communications on Applied Electronics*, vol. 7, núm. 7, pp. 8–13, oct. 2017, doi: 10.5120/cae2017652685.

[26] D. Watson, “Compilers and Interpreters. In A Practical Approach to Compiler Construction ”, *Springer International Publishing*, pp. 13–36, 2017.

[27] J. Gosling y H. McGilton, “The Java language environment. Sun Microsystems Computer Company”, *2550*, 1995.

[28] O. I. Trejos Buritica y L. E. Muñoz Guerrero, “Aprendizaje de la programación con estrategia ‘divide and conquer’ vs. Sin estrategia ‘divide and conquer’”, *Entre ciencia e ingeniería*, vol. 14, núm. 28, pp. 34–39, dic. 2020, doi: 10.31908/19098367.2013.

[29] W. Gonzales Hernandez, “La enseñanza de la informática y de la matemática: ¿semejantes o diferentes?”, *Revista Educación en Ingeniería* , vol. 13, núm. 26, pp. 20–26, 2018.

[30] O. I. Trejos Buriticá, “Aprovechamiento de los tipos de pensamiento matemático en el aprendizaje de la programación funcional”, *Tecnura*, vol. 22, núm. 56, pp. 29–39, abr. 2018, doi: 10.14483/22487638.12807.

[31] M. Felleisen, “How to design program”, *Boston*, 2005.

[32] J. M. Chambers, “Object-Oriented Programming, Functional Programming and R”, *Statistical Science*, vol. 29, núm. 2, may 2014, doi: 10.1214/13-STS452.

[33] P. Hartell y H. Muller, “Functional C. Helsinki”, p. 429, 2006.

[34] O. I. Trejos Buriticá, “Propuesta metodológica para construir un algoritmo que determine si un número es perfecto usando programación imperativa”, *Revista Facultad de Ciencias Básicas*, vol. 13, núm. 2, pp. 83–89, feb. 2017, doi: 10.18359/rfcb.2758.

[35] I. Friss de Kereki, “¿Qué es programar con orientación a objetos? Un enfoque fenomenográfico”, *Cuadernos de Investigación Educativa*, vol. 1, núm. 3, pp. 7–12, mar. 2018, doi: 10.18861/cied.1998.1.3.2804.

[36] L. Careceres Espinoza, “INTRODUCCIÓN A LA PROGRAMACIÓN Introducción, fundamentos de la Programación, herramientas de programación, metodología de Programación, aplicaciones.”, 2019.

[37] A. Gómez y L. Vinces, “COMPARACIÓN DE TENDENCIAS TECNOLÓGICAS EN APLICACIONES WEB.”, 2018.

[38] S. Luján Mora, “Programación de aplicaciones web: historia, principios básicos y clientes web.”, *Alicante, España: Club Universitario*, 2002.

[39] G. Yalupalin, “La programación Front-End y Back-End.”, 2021.

[40] C. Zelaya, “Nuevas tendencias en desarrollo Web.”, *Recuperado de https://www.itcha.edu.sv/publicaciones/ITCHA/1167-2020-12-01/1167-ARTICULO---NUEVAS-TENDENCIAS-EN-DESARROLLO-WEB.pdf*, 2020.

[41] P. Rodriguez, “Las webs y demas plataformas en red. Telefónica Fundación”, 2019.

[42] O. E. Cárdenas Villavicencio, M. P. Zea Ordóñez, M. R. Valarezo Pardo, y R. A. Ramón Ramón, “Comparativa de tendencias de desarrollo de software móvil”, *3C TIC: Cuadernos de desarrollo aplicados a las TIC*, vol. 10, núm. 1, pp. 123–147, mar. 2021, doi: 10.17993/3ctic.2021.101.123-147.

[43] J.-J. Hew, V.-H. Lee, K.-B. Ooi, y J. Wei, “What catalyses mobile apps usage intention: an empirical analysis”, *Industrial Management & Data Systems*, vol. 115, núm. 7, pp. 1269–1291, ago. 2015, doi: 10.1108/IMDS-01-2015-0028.

[44] E. Manrique, “Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo.”, *Revista Ibérica de Sistemas e Tecnologias de Informação.*, 2020.

[45] E. J. Bernal Rozo, R. Cardona Montoya, y J. Páez, “Videojuegos: Avances tecnológicos en aplicación de física e inteligencia artificial”, *Letras ConCiencia TecnoLógica*, pp. 61–78, ago. 2018, doi: 10.55411/26652544.114.

[46] G. Aguilar, “ESTUDIO COMPARATIVO DE LENGUAJES MULTIPLAFORMA JAVA, PHP, QUE PERMITAN GARANTIZAR UN OPTIMO DESEMPEÑO AL DESARROLLAR APLICACIONES DE ESCRITORIO.”, 2022.

[47] S. Salas, *Todo sobre sistemas embebidos*. Editorial UPC, 2015. doi: 10.19083/978-612-318-033-1.

[48] J.-P. Gallo-León, “La biblioteca es servicio (y en ello está nuestro futuro)”, *El Profesional de la Información*, vol. 24, núm. 2, p. 87, mar. 2015, doi: 10.3145/epi.2015.mar.01.

[49] S. Murray, “Bibliotecas. Una historia ilustrada”, 2014.

# ANEXOS