

SEP

SES

TNM

INSTITUTO TECNOLÓGICO DE CHIHUAHUA II



**CURSO ADAPTATIVO MEDIANTE
CONOCIMIENTOS PREVIOS PARA LA MATERIA
DE FUNDAMENTOS DE PROGRAMACIÓN**


**TESIS
PARA OBTENER EL GRADO DE**

MAESTRO EN SISTEMAS COMPUTACIONALES

PRESENTA

GUSTAVO ADOLFO RIVAS HERNÁNDEZ

Leonardo Nevárez Chávez
DIRECTOR DE TESIS
M.C LEONARDO NEVÁREZ
CHÁVEZ


CO-DIRECTOR DE TESIS
M.E.S MARISELA IVETTE
CALDERA FRANCO

CHIHUAHUA, CHIH A 30 DE JUNIO DE 2021.

Dictamen

Chihuahua, Chih., 30 de agosto 2021

**M.C. MARÍA ELENEA MARTÍNEZ CASTELLANOS
COORDINADORA DE POSGRADO E INVESTIGACIÓN
PRESENTE**

Por medio de este conducto el comité tutorial revisor de la tesis para obtención de grado de Maestro en Sistemas Computacionales, que lleva por nombre "CURSO ADAPTATIVO MEDIANTE CONOCIMIENTOS PREVIOS PARA LA MATERIA DE FUNDAMENTOS DE PROGRAMACIÓN", que presenta el C. GUSTAVO ADOLFO RIVAS HERNÁNDEZ, hace de su conocimiento que después de ser revisado ha dictaminado la APROBACIÓN de la misma.

Sin otro particular de momento, queda de Usted.

Atentamente
La Comisión de Revisión de Tesis.



M.C. LEONARDO NEVÁREZ CHÁVEZ

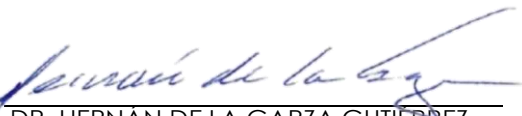
Director de tesis



DRA. MARISELA VETE CALDERA
FRANCO
Co-Director



DR. GREGORIO RONQUILLO MAYNEZ
Revisor



DR. HERNAN DE LA GARZA GUTIERREZ
Revisor

CONTENIDO

ÍNDICE DE FIGURAS.....	iii
ÍNDICE DE TABLAS	iv
1. INTRODUCCIÓN.....	1
1.1 Antecedentes.....	2
1.2 Definición del problema.	4
1.3 Objetivo.....	6
1.3.1 Objetivo general.....	6
1.3.2 Objetivos particulares.....	6
1.4 Justificación.....	6
1.5 Alcances.	8
1.6 Limitaciones.	8
2. ESTADO DEL ARTE	9
3. MARCO TEÓRICO.....	19
3.1 Conceptos.....	19
4. DESARROLLO	37
4.1 Análisis.....	37
4.2 Diseño.....	39
4.3 Programación.....	42
4.4 Pruebas	56
4.5 Implementación.....	59
5. RESULTADOS Y DISCUSIÓN	60
6. CONCLUSIONES	63
7. BIBLIOGRAFÍA.....	65
8. ANEXO A Código fuente.....	68

ÍNDICE DE FIGURAS.

Figura 3.1. Elementos del Aprendizaje adaptativo.....	21
Figura 3.2. Estructura principal de CSS	30
Figura 3.3 Tipos de selectores más comunes en CSS.....	32
Figura 3.4. Modelo basado en cajas en CSS.	32
Figura 4.1. Modelo conceptual.	38
Figura 4.2. Diagrama de secuencia del sistema.....	41
Figura 4.3. Página principal de la plataforma Moodle.	42
Figura 4.4. Curso disponible en la plataforma Moodle.	42
Figura 4.5. Test inicial de conocimientos.....	43
Figura 4.6. Grupos separados después de la clasificación.....	43
Figura 4.7. Mostrar calificaciones del test inicial.....	45
Figura 4.8. Calificaciones del alumno correspondiente a cada examen.....	45
Figura 4.9. Calificaciones correspondientes a cada examen con su Id respectivamente.....	46
Figura 4.10. Grupos creados del curso Fundamentos de Programación.	46
Figura 4.11. Grupos dados de alta en la Base de datos de Moodle.	47
Figura 4.12. Botón para ejecutar la clasificación.	47
Figura 4.13. Alumno perteneciente al Grupo Avanzado.	48
Figura 4.14. Alumnos pertenecientes al Grupo Intermedio.....	48
Figura 4.15. Alumno perteneciente al Grupo Principiante.....	48
Figura 4.16. Lecciones para cada nivel de grupo.	49
Figura 4.17. Fragmento de la lección para el grupo de Principiantes.	50
Figura 4.18. Video explicativo para el grupo de Principiantes.	50
Figura 4.19. Fragmento de lección para el grupo Intermedio.	51
Figura 4.20. Video explicativo para el grupo de Intermedios.	51
Figura 4.21. Fragmento de lección para el grupo Avanzado.....	52
Figura 4.22. Video explicativo para el grupo Avanzado.....	52
Figura 4.23. Restricción de los grupos.	53
Figura 4.24. Restricción para alumnos del grupo Avanzado.....	54
Figura 4.25. Restricción para alumnos del grupo Intermedio.	54
Figura 4.26. Restricción para alumnos del grupo Principiantes.....	55
Figura 4.27. Nombre de alumnos recuperados desde la base de datos.....	56
Figura 4.28. Nombre de los grupos recuperados desde la base de datos.....	56
Figura 4.29. Fragmento de validación de registros.	57
Figura 4.30. Clasificación del alumno a su correspondiente grupo.	58
Figura 5.1. Fragmento del examen de Conceptos Básicos.	60
Figura 5.2. Fragmento del examen Entendimiento de Algoritmos.....	61
Figura 5.3. Fragmento del examen Resolución de algoritmos.	61

ÍNDICE DE TABLAS

Tabla 3.1. Línea de tiempo de Moodle.....	23
Tabla 3.2. Características principales de Moodle.....	24
Tabla 4.1. Caso de uso en un escenario actual.	39
Tabla 4.2. Casos de uso en escenarios futuros.	40
Tabla 4.3. Matriz de conocimientos.	41
Tabla 4.4. Principales tablas de Moodle usadas en el proyecto.	44

RESUMEN: Cada persona tiene una forma particular de aprender, y también el propio ritmo de aprendizaje es diferente. El aprendizaje adaptativo trata de abordar estos aspectos, ya que personaliza y ofrece a los estudiantes recursos que les permitan una forma de aprender que mejor se adapte a las características propias. Además, con la ayuda de los sistemas LMS (Learning Management Systems o Sistemas de Gestión de Aprendizaje) se cuenta con la facilidad de crear cursos en línea, así como recursos didácticos de una forma automatizada, Moodle es uno de los LMS más utilizados. En este proyecto se propone el desarrollo de una herramienta web integrada a la plataforma Moodle para el curso de Fundamentos de Programación, dicha herramienta permitirá incluir características adaptativas al curso, considerado el criterio de los conocimientos previos de los estudiantes. Los conocimientos previos se evaluarán por medio de un cuestionario al inicio del curso. El cuestionario está diseñado para determinar los conocimientos básicos en teoría, entendimiento y resolución de algoritmos. De esta manera el sistema proveerá a cada estudiante con los recursos didácticos que sean más adecuados para él, buscando un mejor nivel de desempeño académico.

1. INTRODUCCIÓN

1.1 Introducción.

El aprendizaje adaptativo es un método educativo que modifica sus contenidos y formas de enseñar de acuerdo a cada alumno. Para ello recolecta información sobre los hábitos de aprendizaje, conocimientos, debilidades y fortalezas de cada usuario, para crear un plan de estudios a la medida. Se hace énfasis en las áreas donde los alumnos tienen más dificultades y se adaptan a la forma de aprendizaje de cada uno, diseñando un camino de enseñanza personalizado y adaptativo único para cada estudiante (CEDDIE, 2015).

Según García (2013) hace referencia a que ningún par de estudiantes es idéntico, cada uno aprende a su propio ritmo, provienen de un proceso de desarrollo pedagógico diferente, pertenecen a un contexto cultural específico, tiene motivaciones e intereses diferentes, tienen ritmos de atención diferentes y modos de aprender particulares.

El aprendizaje adaptativo hace que se genere un contenido dinámico e interactivo, y coloca al estudiante en el centro de su experiencia de aprendizaje individual. Las herramientas de aprendizaje adaptativo ayudan a supervisar la interrelación del estudiante con el sistema, así como su aprendizaje.

El proyecto se desarrolló en el Tecnológico Nacional de México, Instituto Tecnológico de Chihuahua II, ubicado en la Avenida las Industrias #11101 Complejo Industrial Chihuahua, Chihuahua, Chih. Fundada el 14 de septiembre de 1987, actualmente la institución cuenta con siete carreras disponibles, las cuales son Ingeniería Informática, Ingeniería en Diseño Industrial, Licenciatura en Administración, Arquitectura, Ingeniería en Gestión Empresarial, Ingeniería Industrial e Ingeniería en Sistemas Computacionales (I.S.C). Principalmente la carrera de Ingeniería en Sistemas Computacionales cuenta con las materias de programación que atiende desde Fundamentos de Programación hasta Programación de Dispositivos Móviles.

El objetivo de este proyecto es implementar una herramienta web con el fin de que los estudiantes puedan aprender y/o reforzar sus conocimientos previos en la materia de Fundamentos de Programación mediante Moodle, ya que es un LMS de código abierto de gran versatilidad, pues permite generar cursos con muy diferentes configuraciones, que les provea herramientas de aprendizaje, con la ayuda de exámenes que ayuden a evaluar las dimensiones de conocimiento teórico/práctico. La integración de técnicas de programación ayudará a clasificar a los alumnos en grupos con base a su nivel de conocimientos y a su vez se le proporcionará material de apoyo para la comprensión de los temas de la materia.

1.1 Antecedentes.

El Instituto Tecnológico de Chihuahua II cuenta con una plataforma de aprendizaje llamada Moodle, en la cual los alumnos consultan los recursos educativos que el profesor sube a la plataforma.

Moodle es una plataforma de aprendizaje diseñada para proporcionar a los administradores, profesores y estudiantes, un sistema integrado para crear ambientes de aprendizaje personalizado (Moodle, 2018).

Según López (2015) las posibilidades adaptativas de Moodle aparecen esencialmente en dos direcciones. Se podría decir que una es intra-actividad y otra es inter-actividades. La primera significa que la adaptación al usuario es una posibilidad de la configuración interna de la actividad; mientras que las inter-actividades supone el establecimiento de reglas o relaciones de dependencia entre diferentes actividades. Algunas actividades de Moodle tienen cierta capacidad adaptativa propia, como objetos de aprendizaje independientes del resto de recursos o actividades de un curso Moodle. Por ejemplo, la adaptabilidad intra-actividad está disponible en los módulos de lecciones, cuestionarios, foros y ScoRM (acrónimo de Sharable Content Object Reference Model).

A continuación, se describen las modalidades utilizadas en el proyecto que ofrecen adaptabilidad:

- Lecciones de Moodle: Son actividades de aprendizaje que proporcionan contenidos de forma individualizada siguiendo una programación ramificada. Las lecciones están formadas por páginas puramente de contenidos o con una pregunta. En las primeras, se ofrece al estudiante la posibilidad de elegir la siguiente página que quiere ver y, en las segundas, la respuesta dada a la pregunta que contiene determina la próxima página que el estudiante va a ver. La navegación en una lección puede ser lineal o tan compleja como el formador estime necesario
- Los cuestionarios de Moodle: Son actividades en las que se ofrece al estudiante una lista de preguntas cuya respuesta conduce a una calificación. La adaptabilidad de un cuestionario se establece al configurar la visualización de la retroalimentación incluida en las preguntas y sus opciones de respuesta, así como al seleccionar el comportamiento del cuestionario con interactividad para que muestre, si fuera necesario, las pistas añadidas a las preguntas. Se trata, de una realimentación individualizada, inmediata y automática.

Edmodo es una plataforma social educativa gratuita que permite la comunicación entre los alumnos y los profesores en un entorno cerrado y privado a modo de microblogging, permite crear clases dinámicas, estrategias de comunicación remota entre estudiantes y docentes y transmitir lecciones en vivo usando Edmodo con YouTube u otras plataformas de vídeo.

La plataforma admite tres tipos diferentes de perfiles o roles. En primer lugar, se encuentran los profesores, aquellos que crean y administran los grupos, pueden programar los eventos, asignar calificaciones a los alumnos, dar clases en streaming o subir el material de sus aulas.

En segundo lugar, se encuentran los alumnos, que tienen que utilizar un código de acceso para unirse a los grupos. Una vez dentro pueden consultar las clases, descargar archivos, acceder a sus notas e interactuar con el profesor. También participar exponiendo dudas o debatiendo con compañeros, al igual que sucede en los grupos en redes sociales como Facebook (Núñez-Torrón, 2021).

Blackboard es una plataforma que integra un ambiente sólido de enseñanza y aprendizaje en línea. Se caracteriza por administrar un conjunto de recursos que permiten desarrollar cursos

virtuales, específicamente: impartir y distribuir contenidos que se encuentran presentados en diversos formatos (texto, sonido, video y animación), realizar evaluaciones en línea, llevar a cabo el seguimiento académico de los alumnos participantes, asignar tareas y desarrollar actividades en ambientes colaborativos.

Blackboard fue diseñado para instituciones dedicadas a la enseñanza y el aprendizaje y proporciona la funcionalidad necesaria para administrar correctamente programas de educación a distancia o a través de la Web. Blackboard es una plataforma computacional flexible, sencilla y de uso intuitivo, que ofrece herramientas de creación de cursos y contenidos, una nueva manera de evaluación, herramientas de colaboración síncronas y asíncronas y administración académica por parte de profesores.

Blackboard cuenta con funciones que facilitan el "e-learning" permitiendo el acceso a un formato de portal desde el cual se cuenta con toda la información detallada de cursos. Además de esto, el docente cuenta con la integración de sistemas basados en Web que permiten crear el diseño de las actividades del curso, personalizar el formato de diversos documentos, utilizar herramientas para el trabajo colaborativo, realizar evaluaciones mediante la plataforma, proporcionar diversos recursos de apoyo al curso y permite acceder a diferentes cursos desde un portal y cuenta únicos (EcuRed, 2016).

1.2 Definición del problema.

Cuando los alumnos ingresan a la institución, específicamente a la carrera de Ingeniería en Sistemas Computacionales, generalmente no cuentan con fundamentos previos sobre el conocimiento de la programación, independientemente del lenguaje que sea. Por ello en ocasiones les resulta difícil aprender o adquirir conocimientos nuevos. Sin embargo, también se cuenta con estudiantes, con base a su formación académica, que cuentan con conocimientos en el área de la programación. Los sistemas basados en LMS pueden ayudar a los usuarios a reforzar sus conocimientos de una manera intuitiva y personalizada.

Las clases están llenas de alumnos únicos que aprenden de forma diferente, tienen experiencias educativas previas distintas, y cuentan con necesidades, niveles de atención e intereses individuales. Es difícil para un profesor saber exactamente en qué nivel se encuentra cada estudiante, dónde tiene dificultades y por qué.

Teniendo esto en cuenta, la mayoría de los cursos están enfocados para todos los alumnos en general y se pretende que todos avancen de la misma manera, pero esto en muchos casos no es posible, ya que como se ha mencionado los estudiantes aprenden de diferente manera.

El aprendizaje adaptativo a través de herramientas en línea no limita el trabajo del profesor, ni elimina el contacto con el estudiante. Al contrario, facilita la labor docente, agiliza el aprendizaje, le da información útil sobre cómo aprenden sus alumnos y ahorra tiempo para realizar otras actividades y trabajar de forma más directa con los estudiantes, aquellos aspectos que suponen un reto mayor para ellos.

Los docentes, mejoran el conocimiento de los alumnos y de sus capacidades, puntos débiles y fortalezas en el aprendizaje, y pueden dirigirse a ellos, de manera más individualizada y adecuada a sus objetivos. Así, pueden focalizar su atención en los aspectos menos consolidados, abordar las dificultades de forma personalizada o plantear nuevos retos, a los alumnos más avanzados para que no pierdan el interés.

Actualmente, no se cuenta con un sistema para clasificar a los alumnos de manera automatizada según sus conocimientos sobre la materia de Fundamentos de Programación, ya que por lo general se dan de alta en el grupo correspondiente a la materia y el docente se encarga de dar la clase a todos por igual sin saber el nivel de conocimientos con los que cuenta cada alumno.

1.3 Objetivo.

1.3.1 Objetivo general.

Implementar un curso adaptativo basado en los conocimientos previos para mejorar el nivel de aprendizaje de los alumnos para la materia de “Fundamentos de Programación”, este sistema analizará los conocimientos que tenga el alumno al inicio del curso. Mediante las técnicas de programación, se obtendrán los resultados de exámenes y se le asignará al alumno un grupo donde tendrá recursos educativos según su nivel.

1.3.2 Objetivos particulares

- Creación de un sistema que permitirá con base a los resultados obtenidos clasificar a los alumnos en conjuntos con las mismas características.
- El curso contendrá tres exámenes: conceptos básicos, entendimiento de algoritmos y resolución de algoritmos.
- El sistema asignará a cada estudiante un Grupo de Moodle automáticamente.
- El sistema le proporcionará según su nivel de conocimientos previos lecturas, videos y cualquier contenido informativo que le ayude con su aprendizaje.

1.4 Justificación.

Existen diversas herramientas LMS enfocadas a la educación, como aquellas para las materias de matemáticas, física, química, etc. Hay pocos recursos en línea diseñados especialmente para el aprendizaje de la programación, la mayoría tiene costos elevados y en ocasiones no resulta sencillo su diseño, y la facilidad de uso no es la adecuada tanto para el alumno como para el profesor que utilice el sistema.

Según Lozano (2016), las nuevas tecnologías y las plataformas digitales son una herramienta esencial para aplicar el aprendizaje adaptativo en el aula de manera eficaz, ya que permiten recopilar, analizar y evaluar las actividades, trabajos, acciones o tareas del alumno, y

monitorizar gran cantidad de datos con precisión y rapidez para plantear automáticamente nuevas propuestas.

Estos sistemas de trabajo en línea combinan los datos recopilados de los alumnos (con multiplicidad de parámetros cuantitativos y cualitativos) con información estadística, respuestas predefinidas, patrones de aprendizaje ya establecidos y algoritmos especialmente diseñados para procesar estos datos e identificar fortalezas y debilidades. No solo se tienen en cuenta las respuestas, sino también la forma de interactuar del estudiante o el tiempo que invierte en completar una acción o sus dudas. Con ello, la plataforma crea una ruta de aprendizaje específica para el alumno. Además, al ser plataformas en línea y digitales, permiten contrastar datos de estudiantes que ayudan a generar nuevas pautas, de forma que el propio sistema va mejorando sus características y ajustando sus resultados cuanto más se utiliza (retroalimentación) (Lozano, 2016).

El aprendizaje adaptativo es una herramienta más con la que pueden contar los docentes. Puede aplicarse a una o varias materias, y complementarse con otros instrumentos TIC. El profesor puede programar un tema y explicarlo o empezar a trabajar un proyecto. Al mismo tiempo puede programar ese tema en la plataforma de aprendizaje adaptativo y, conforme los alumnos avancen, el sistema inteligente detectará las dificultades y generará los distintos itinerarios de aprendizaje para cada alumno. La información suministrada por el sistema permitirá al docente detectar qué temas presentan dificultades o a qué alumnos les cuesta avanzar. En el aula podrá trabajar esos conceptos o habilidades de forma específica, con un solo alumno o un grupo de ellos, convirtiendo el tiempo en el aula en un espacio de interacción y guía mucho más rico y eficaz (Lozano, 2016).

1.5 Alcances.

- La adaptación incluye como criterio el conocimiento previo, se aplica a una sola materia (Fundamentos de Programación) del instituto.
- Al iniciar el curso se aplicará un examen para la evaluación de los conocimientos previos con que el alumno cuente.
- El programa de estudios abarcará el curso por completo, pero las pruebas se realizarán a un solo tema.

1.6 Limitaciones.

- La procedencia de cada alumno, ya que existen instituciones donde no imparten materias de programación.
- El tiempo para desarrollo del sistema es muy corto para abarcar todo el programa de estudio.
- La infraestructura, como la red de internet, con la que cuenta la institución puede afectar el funcionamiento a la hora de ejecutar el software.

2. ESTADO DEL ARTE

A continuación, se describe algunos sistemas adaptativos que se han desarrollado a lo largo de la historia de la computación, para entrar en contexto con lo relacionado en este proyecto.

1905: Primer examen adaptativo Binet IQ Test

Según Guirao (2011), el test de Binet-Simon es un test de inteligencia creado por el psicólogo francés Alfred Binet y por el psiquiatra de la misma nacionalidad Théodore Simon en 1905, por encargo del Ministerio de Educación de su país.

Con este primer test de inteligencia se trataba de determinar la inteligencia de los individuos que presentaban déficit intelectual para ver sus posibilidades educativas.

Binet y Simon idearon atribuir a cada sujeto una edad mental, al margen de su edad cronológica. La edad mental no hacía referencia a los años que tenía la persona, sino a lo que el individuo examinado era capaz de hacer en relación con lo que lograban en una prueba concreta el resto de la población media de una determinada edad cronológica

Por lo tanto, en este test de inteligencia cuando una persona era capaz de realizar con éxito todas las pruebas correspondientes a una edad dada, ésta se consideraba como su “edad mental base”. Si continuaba superando las siguientes, se le debía ir añadiendo una fracción de año. Cuando fracasaba en todas las pruebas de una determinada edad el test se detenía.

Algunas de las pruebas eran como las siguientes:

- A los tres años las preguntas para determinar la edad mental base eran decir el apellido, repetir dos cifras, describir un grabado y repetir una frase corta. Y como complemento debía señalar la nariz, el ojo y la boca.
- A los cuatro años tenía que repetir tres cifras, como preguntas complementarias, decir su sexo, nombrar llave, cuchillo, moneda, y comparar dos líneas.
- A los cinco años se les decía una frase larga que debían repetir y contar cuatro monedas sencillas. Las actividades complementarias consistían en comparar dos pesos, copiar un cuadro y realizar un rompecabezas.

1954: Máquina de Skinner

B. F. Skinner en 1954 decidió interesarse por aplicar los principios del aprendizaje a la educación ya que vio que los métodos que se aplicaban en el ámbito escolar eran totalmente contrarios a los principios de aprendizaje. Un buen día decidió acudir a una clase de matemáticas, en la cual se encontraba su hija, y observó cómo el docente mandaba a sus alumnos realizar un número de cuentas y problemas sin recibir ningún tipo de retroalimentación, por lo que Skinner se dio cuenta de que ése no era un método adecuado para que los niños aprendieran matemáticas, así que decidió, entre otras cosas, crear la llamada máquina de enseñanza, con una enseñanza programada.

Esta máquina consistía en una caja en la que el profesor introducía una hoja con todos los conceptos que el alumno tenía que aprender y otra lámina donde se podía ocultar parte del texto. El procedimiento era el siguiente: el estudiante iba leyendo el texto introducido en la caja e iba escribiendo cada una de las respuestas a las preguntas que le iban apareciendo. A continuación, el alumno rodaba la máquina y si estaban respondidas correctamente las preguntas, la hoja con las preguntas iba pasando y se anotaba un punto; de lo contrario la máquina no dejaba avanzar y el alumno debía volver atrás, obligándole a volver a leer el texto de nuevo.

Esta máquina mecánica no fue creada como elemento único de aprendizaje, sino como un apoyo al profesor. La máquina no enseñaba por sí misma, sino que ponía en contacto al alumno con el material de una forma estructurada, y le ayudaba en el aprendizaje progresivo de conceptos más complejos. También presentaba las tareas para aprender en pequeños pasos progresivos, diseñados en una secuencia programada de lo más fácil a lo más difícil (Martín, 2012).

1959: PLATO (Programmed Logic for Automated Teaching Operations).

Se considera como un sistema CAI (Computer Assisted Instruction) temprano. Se inició en la Universidad de Illinois, posteriormente desarrollada por Control Data Cooperation. El objetivo de este sistema de aprendizaje basado en computadora, es automatizar la instrucción individual, durante los posteriores siete años se ha examinado la utilidad y la viabilidad del sistema de enseñanza por computadora. Durante esos años, aproximadamente fueron desarrollados 300 programas para demostrar su viabilidad para la enseñanza y la investigación educativa. Las áreas temáticas de programas PLATO eran muy amplias desde la adquisición de lenguas extranjeras, matemáticas, ciencias para primaria, secundaria, educación superior y en el lugar de trabajo (Won, 2016).

1970: SCHOLAR: Primer sistema tutorial inteligente.

En 1970 se desarrolló el primer STI en el Instituto Tecnológico de Massachussets (MIT): SCHOLAR. Este sistema estuvo basado en una aplicación conocida como ELIZA diseñada para el estudio de la comunicación entre persona-máquina. SCHOLAR se diseñó para enseñar geografía del continente americano y supuso un salto cualitativo sobre los sistemas educativos anteriores. Previamente a 1970, el proceso de enseñanza se basaba en series de preguntas y respuestas pre-diseñadas que hacían el sistema rígido, mientras que en SCHOLAR el formato de representación de la información permitía al sistema anticipar las respuestas de la persona que aprende y adaptarse a sus necesidades (Lozano, 2016).

1980: Administración de exámenes por computadora

La década de los 80 puede ser conocida en este ámbito como la década en que se produce el desarrollo de test informatizados siguiendo los criterios de los test convencionales y se inicia la evaluación adaptativa informatizada (Lozano, 2016).

- El Educational Testing Service desarrolló sistemas multimedia para pruebas de certificación y nivel.

- El Waterford Testing Center realizó el diseño de un conjunto de test informatizados para evaluar los conocimientos de alumnos de primaria en Lectura, Lenguaje y Matemáticas.
- McDermott desarrolló un conjunto de programas para evaluar los posibles problemas intelectuales, motivacionales o conductuales que pueden estar en la base de determinados problemas de aprendizaje.
- Kent y Albanese realizan en 1987 la primera evaluación con test adaptativos informatizados de estudiantes de medicina de segundo curso.
- En 1988 se comienza la introducción paulatina de exámenes mediante TAI (Tests Adaptativos Informatizados) en institutos de Estados Unidos.
- El laboratorio Armstrong de las fuerzas armadas norteamericanas desarrolla la batería de test informatizados Cognitive Abilities Measurement (CAM), con la que se intentan medir habilidades de procesamiento de la información que anteriormente se estudiaban en los laboratorios experimentales.

1988: Primera conferencia internacional de ITS (Sistemas Tutoriales Inteligentes)

En este año, se efectuó la primera conferencia internacional de ITS donde todos los investigadores en este campo tuvieron la oportunidad de conocer, aprender, compartir conocimientos y enterarse de los logros que se han obtenido en esta área alrededor del mundo (Lozano, 2016).

1990: Computer Assisted Instruction CAI

Es a partir de este año cuando se produce un auge en el uso de los CAI (es un tipo de programa educativo diseñado para servir como herramienta de aprendizaje) debido a que el uso de la computadora empieza a ser más generalizado debido a que se han reducido costes, es más accesible (Lozano, 2016).

1993: AiED (Artificial Intelligence in Education).

Tuvo lugar la primera conferencia internacional de inteligencia artificial en educación (AiED) en Edimburgo, Reino Unido. Suelen estar compuestas por una serie de conferencias de alta calidad sobre las últimas investigaciones en sistemas inteligentes y ciencias cognitivas

enfocadas a las aplicaciones informáticas relacionadas con el ámbito de la educación y de todos los campos que componen la AiED, tales como la informática, las ciencias del aprendizaje, diseño de juegos, psicología, sociología, lingüística, educación, etc. (Romera, 2015).

1998: Carnegie Learning

Se trata de un proveedor líder en planes de estudio de matemáticas para los grados de 6 a 12. Las soluciones de Carnegie de aprendizaje incluyen los libros de texto de matemáticas con actividades de clase, de colaboración centradas en el estudiante, junto con un software innovador, adaptable y desarrollo profesional docente (Lozano, 2016).

2000: LMS (Learning Management System)

Los sistemas de gestión del aprendizaje (LMS) tienen una historia diferente. El sistema de aprendizaje integrado (ILS), que ofrece una funcionalidad adicional más allá del contenido instruccional, una instrucción más personalizada y la integración con todo el sistema. El término ILS fue acuñado originalmente por Jostens y el LMS fue utilizado originalmente para describir la parte del sistema de gestión del método de aprendizaje PLATO K-12, libre de contenido y separado del material didáctico. El término LMS se utiliza actualmente para describir una serie de aplicaciones informáticas educativas.

Las principales funciones del sistema de gestión de aprendizaje son: gestionar usuarios, recursos, así como materiales y actividades de formación, administrar el acceso, controlar y hacer seguimiento del proceso de aprendizaje, realizar evaluaciones, generar informes, gestionar servicios de comunicación como foros de discusión, videoconferencias, entre otros (Lozano, 2016).

2008: KNEWTON

Es una compañía de aprendizaje adaptativo que ha desarrollado una plataforma para personalizar el contenido educativo. Esta plataforma permite a colegios, editoriales y desarrolladores ofrecer el aprendizaje adaptativo para cualquier estudiante.

Se trata de un proveedor que hace posible que otros puedan construir aplicaciones con este método de aprendizaje. Esta tecnología permite a la empresa llevar a cabo sofisticados análisis en tiempo real de grandes cantidades de datos del desempeño de los estudiantes. Knewton utiliza la tecnología de aprendizaje adaptativo para identificar las fortalezas y debilidades de cada estudiante. Los conceptos están etiquetados en cada nivel, lo que permite a la plataforma hacer recomendaciones personalizadas basadas en las capacidades y necesidades de los estudiantes (Lozano, 2016).

2010: SMART SPARROW

Las herramientas de software de Smart Sparrow, conocidas colectivamente como Adaptive eLearning Platform, son un conjunto basado en la web que desarrolla contenido y aplicaciones de aprendizaje adaptativo, despliega ese material a los estudiantes y analiza cómo los estudiantes aprenden de sus respuestas al material.

La plataforma implementa un sistema de tutoría inteligente que combina Modelado basado en Restricciones con Rastreo de Modelos.

El aprendizaje adaptativo es un método de instrucción que utiliza un sistema por computadora para crear una experiencia personalizada de aprendizaje. Bajo esta premisa se sabe que este tipo de aprendizaje no es algo de la última década, sino que su origen normalmente se relaciona con la máquina Skinner y a la Teoría del Aprendizaje Programado que surgieron en la década de 1950 y continuó con el movimiento de la Inteligencia Artificial en la década de 1970. En el aprendizaje adaptativo, la premisa básica es que la herramienta o el sistema será capaz de adaptarse al método de aprendizaje del estudiante, lo que resulta en una mejor y más eficaz experiencia de aprendizaje para el usuario. Ya en los años 70 la principal barrera fue el coste y el tamaño de los equipos, lo que hizo impracticable la aplicación generalizada. Otro obstáculo en la adopción de sistemas inteligentes iniciales era que las interfaces de usuario no eran propicias para el proceso de aprendizaje (Lozano, 2016).

Con la evolución de las Tecnologías de la Información y de la Comunicación (TIC) y gracias a que los dispositivos móviles que se han vuelto más pequeños, potentes y menos costosos, el aprendizaje adaptativo es ahora aplicable a la enseñanza en el aula, a distancia y en los

escenarios de tutoría. Hoy en día los sistemas de aprendizaje adaptativo ya se están utilizando en una gran variedad de entornos para enseñar y entrenar de manera más eficaz, por ejemplo: la NASA y diversas áreas militares de Estados Unidos los están utilizando en sus programas; Amazon y Netflix, también han adoptado esta tecnología para anticipar las preferencias de sus clientes (Lozano, 2016).

2011: MOOC

Es el acrónimo en inglés de Massive Online Open Courses (o Cursos online masivos y abiertos) Es decir, se trata de un curso a distancia, accesible por internet al que se puede apuntar cualquier persona y prácticamente no tiene límite de participantes.

Se puede considerar a David Wiley como el autor del primer MOOC conceptual, iniciado en la Universidad de Utah en agosto de 2007. Se trataba de un curso de educación abierta. Esta iniciativa tuvo continuidad en numerosos proyectos impulsados desde diferentes centros universitarios dentro y fuera de los Estados Unidos (Lozano, 2016).

2011: SI-APRENDE.

El Sistema Inteligente SI-APRENDE utiliza un modelo tutor partiendo del modelo de un Sistema tutorial inteligente (ITS) para establecer una secuencia y navegación adaptativas de objetos de aprendizaje, bajo el estándar SCORM.

El modelo del ITS se basa en redes de decisión dinámicas para seleccionar la acción pedagógica que mejor se adapte a la situación de aprendizaje de cada estudiante. Fue desarrollado en colaboración con el Instituto de Investigaciones Eléctricas como parte de un proyecto de tesis de maestría (Argotte, 2011).

Aaprender.

Este proyecto propone el desarrollo de una herramienta adaptativa de aprendizaje en línea llamada Aaprender. Esta herramienta es un sistema en el cual cada estudiante sigue una secuencia de ejercicios distinta, dependiendo de su interacción con el sistema, y proporciona retroalimentación a los estudiantes de acuerdo a sus respuestas anteriores. Si la opción elegida es la correcta, el sistema les proporciona un problema reto de mayor nivel de

dificultad o les despliega el siguiente problema de la lista; si la opción seleccionada es incorrecta, le proporciona un problema derivado del mismo tipo, aunque de menor nivel de dificultad para remediar ese error particular. Adicionalmente, los profesores pueden generar reportes específicos sobre el desempeño de los estudiantes y del grupo en general.

En este proyecto se desarrolla la construcción de un módulo llamado **“Modelo de Navegación adaptativa sobre la plataforma Moodle”** que posibilite la adaptabilidad de navegación y presentación de los contenidos para cursos desarrollados sobre la plataforma Moodle, al progreso y preferencias de aprendizaje de los estudiantes. Lo que se busca es definir una estructura de curso mucho más flexible que pueda adaptarse mejor a las exigencias, necesidades y preferencias de cada estudiante. Esto es mediante la construcción de un modelo del estudiante que incorpore la información que se obtiene de cuestionarios, registros de interacción con el sistema, comentarios y demás. Una estructura de curso enriquecida con información que provea al usuario con herramientas que le permitan posicionarse dentro de la estructura del curso y determinar cuál es la mejor estrategia a seguir (García, 2013).

En el 2014 se desarrolló un proyecto llamado **“Diseño de un aprendizaje adaptado a las necesidades del alumno”** en la universidad de Salamanca en España, donde las clases pueden ser presenciales o virtuales o viceversa. Este modelo da el protagonismo del aprendizaje al alumno, siendo el profesor el coordinador de su actividad y tutor de sus necesidades académicas. Se utilizó d-learning es un medio de formarse que integra la enseñanza presencial con la virtual de forma síncrona: el alumno puede seguir la misma semana, incluso el mismo día, el mismo contenido, de forma virtual y de manera presencial, aunque no tiene por qué llevar el mismo ritmo. Tras tres años de aplicación del modelo, las asignaturas participantes han mejorado la tasa de rendimiento en más del 70% y la tasa de éxito en más del 35% (Romero, 2014).

En el 2016 en la universidad de Murcia España se desarrolló un modelo llamado **MeL: Modelo de adaptación dinámica del proceso de aprendizaje en eLearning**. En este trabajo se describe un modelo de adaptación para Learning Management Systems (LMS) que utilizando variables centrales en el proceso de aprendizaje permite aplicar reglas adaptativas

a los distintos tipos de contenidos y conocimientos que se han de transmitir o adquirir, tales como la autoeficacia y las metas de estudio. A nivel aplicado, el modelo obtenido permite desarrollar cursos adaptados que dan soporte y promueven el aprendizaje y la autorregulación dentro de los entornos de aprendizaje virtuales (Sánchez, 2016).

Promoción de la autogestión a través de objetos de aprendizaje adaptativos en alumnos de educación superior (2018).

El propósito de este trabajo fue evaluar la relación del uso del sistema de objetos de aprendizaje adaptativos Kavelearning como herramienta didáctica digital en la enseñanza y en el desarrollo del aprendizaje autónomo en la asignatura de Estructura de datos de la carrera de Ingeniería en Computación.

El desarrollo consistió en el diseño de una estrategia didáctica mediante la prueba piloto del sistema Kavelearning con el fin promover el aprendizaje autogestivo en los estudiantes a través de objetos de aprendizajes adaptativos. Dicho sistema se implementó en la página www.Kavelearning.com a través de la cual fueron desarrollados diferentes objetos adaptativos para la asignatura de “Estructura de datos” de la carrera de Ingeniería en Computación durante el calendario escolar de agosto a diciembre de 2017 (calendario B), a un grupo de 41 alumnos, de los cuales el 93% son hombres y el 7% mujeres.

El diseño de la estrategia didáctica partió del principio del aula invertida donde primeramente el estudiante accedía al sistema Kavelearning para consultar el tema previo a la sesión presencial, para después realizar un conjunto de actividades prácticas en el salón de clases. Finalmente, en el módulo de auto evaluación, el estudiante valoraba su avance y necesidades de aprendizaje.

El sistema Kavelearning está creado a partir de los siguientes módulos: Detección del estilo de aprendizaje, examen diagnóstico, presentación de contenidos mediante objetos de aprendizaje, finalmente la auto evaluación del aprendizaje, con la finalidad de medir el avance de competencias específicas de la asignatura de Estructura de datos (Fernández, 2018).

RAIM: Framework para la inclusión adaptativa en entornos educativos para todos. (2020).

Se desarrolló una plataforma tecnológica que integra herramientas para la creación, etiquetado, almacenamiento, publicación, búsqueda, recuperación y adaptación de recursos educativos (Tabares-Morales, 2020).

La plataforma desarrollada fue denominada RAIM y cuenta con usuarios docentes, a quienes se les presentan funcionalidades asociadas con la gestión de los recursos educativos, y usuarios estudiantes, a quienes se les entrega el material educativo de forma adaptada.

Se diseñó un formulario que permite identificar en cinco pasos las características, necesidades y preferencias de los usuarios estudiantes. Además, a través de la plataforma, se hace seguimiento a las interacciones del usuario, lo que permite hacer algunas actualizaciones de su perfil.

A continuación, se describe brevemente cada una de las herramientas disponibles mediante la plataforma RAIM:

- EduTools: Esta herramienta de autor permite la creación de actividades educativas en HTML, con un editor que facilite el proceso de construcción para usuarios con pocos conocimientos técnicos. Se pueden crear tres tipos de actividades diferentes: libro, contenido web y cuestionario.
- GAIATools: Es una herramienta de autor para el desarrollo de objetos de aprendizaje orientados a estudiantes con discapacidad visual y auditiva, ya que ofrece mecanismos que facilitan la interacción con esta población. Ofrece la posibilidad de creación de cinco tipos diferentes de recursos, entre los que se incluyen, objetos de aprendizaje textuales, interactivos y tipo libro, entre otros.
- ARTools: Esta herramienta es una novedosa propuesta que facilita la creación de material educativo gracias a tecnologías de realidad aumentada. Su utilización es posible sin la necesidad de que el docente tenga previo conocimiento de esta tecnología; así, genera recursos que permiten el aprendizaje activo de los estudiantes.

3. MARCO TEÓRICO

En este capítulo, se hace una descripción de conceptos, herramientas, lenguajes y tecnologías que se aplican para el desarrollo del proyecto.

3.1 Conceptos

Para entender un poco más sobre los conceptos relacionados en este proyecto, a continuación, se describirán cada uno de ellos:

Ingeniería de software.

La ingeniería del software es el desarrollo, operación y mantenimiento del software de forma sistemática, disciplinada y cuantificable, y el estudio de dichos métodos. En otras palabras, es el estudio dedicado a la creación de software de buena calidad, barato y fácil de desarrollar y mantener. Es la aplicación de la ingeniería al software.

La ingeniería del software comienza a formalizarse a finales de la década del 1960. Con el transcurso de los años se han desarrollado recursos que conforman la ingeniería del software, es decir, herramientas y técnicas de especificación, diseño e implementación del software.

La utilización de determinados recursos depende de la magnitud del proyecto, de la empresa a cargo, la experiencia de los desarrolladores, el financiamiento con el que se cuenta, etc.

Subdisciplinas de la ingeniería del software:

- **Requerimientos del software:** la obtención, análisis, especificación y validación de los requerimientos para el software.
- **Diseño de software:** diseño del software, generalmente llevado a cabo con herramientas CASE y el uso de lenguajes de diseño como el UML (lenguaje unificado de modelado).
- **Desarrollo/implementación de software:** es la construcción del software empleando lenguajes de programación.
- **Testeo/prueba del software.**

- Mantenimiento del software: solución de los problemas del software ya desarrollado.
- Gestión de la configuración del software: debido al aumento de la complejidad del software, su configuración debe ser gestionada usando métodos estandarizados y estructurados.
- Gestión de la ingeniería del software.
- Proceso de desarrollo del software: es el procedimiento de la construcción del software. Sus dos paradigmas más importantes son: modelo de desarrollo en cascada y modelo de desarrollo ágil de software.
- Herramientas de ingeniería del software: como CASE.

Aprendizaje Adaptativo.

Básicamente el aprendizaje adaptativo es una estrategia de diseño instruccional que personaliza las actividades que se le proponen a cada participante, sobre la base de su desempeño previo. En la práctica, la experiencia de aprendizaje comienza con una evaluación que determina las áreas en las que el usuario es competente y las que requieren de mayor trabajo. Luego, utilizando esta información, se pasa al momento instruccional en el que se le presentan al aprendiz, únicamente las actividades que contribuyen al logro de sus áreas débiles. Así, los participantes no tienen que realizar todas las actividades o trabajar todos los contenidos de un curso, ni siquiera todos tienen que revisar la misma información, sino que a cada uno se le presenta solo la información que realmente necesita (Llano, 2019).

Los beneficios de utilizar el aprendizaje adaptativo son:

- Al comenzar con la evaluación, el participante percibe que se le está tomando en cuenta, y que se le ofrecen contenidos y actividades ajustados específicamente a su perfil.
- Con la información anterior, se genera una experiencia que se concentra solamente en los contenidos que son significativos para el aprendiz, obviando los que le son irrelevantes.

- Y finalmente, esto reduce el tiempo necesario para tomar el curso, ya que solo debe trabajarse con lo que se necesita.

En la figura 3.1 se puede observar los principales elementos que conforman el aprendizaje adaptativo.

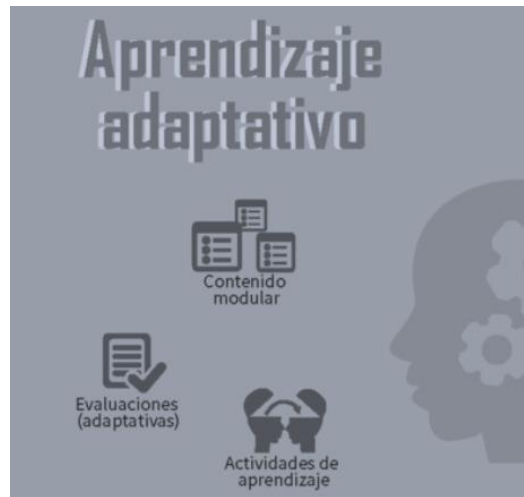


Figura 3.1. Elementos del Aprendizaje adaptativo. (Lozano, 2016)

LMS

Los sistemas LMS son espacios virtuales de aprendizaje que están orientados a facilitar la experiencia de capacitación a distancia, ya sea para empresas o para instituciones educativas. El sistema permite la creación de aulas virtuales donde se produce la interacción entre profesores y alumnos, también se pueden hacer evaluaciones, intercambiar archivos, participar en chats o foros, entre otras cosas. A continuación, se presentan algunas características y beneficios de las plataformas LMS (UIV, 2018).

Las características de las plataformas LMS pueden sintetizarse en los siguientes puntos:

- Estandarización: Permite utilizar cursos realizados por terceros, personalizando el contenido y reutilizando el conocimiento.
- Flexibilidad: La plataforma puede ser adaptada tanto a planes de estudio de una institución, como para el uso para capacitación de una empresa.

- **Interactividad:** Permite ser protagonistas de nuestro propio aprendizaje a través del autoservicio y servicios guiados.

Entre los beneficios que tienen las plataformas LMS según la Universidad Internacional de Valencia (2018) se destacan las siguientes:

- Permiten estudiar en cualquier momento y en cualquier lugar, quitando el problema de las distancias geográficas o temporales, ofreciendo una inmensa libertad en cuanto a tiempo y ritmo de aprendizaje.
- Para utilizar una plataforma LMS no es necesario tener grandes conocimientos de computación, solo basta con un nivel básico del funcionamiento del Internet y de las herramientas informáticas.
- Posibilitan la capacitación de las personas con máxima flexibilidad y a costos reducidos.
- Posibilitan un aprendizaje constante y actualizado a través de la interacción entre tutores y alumnos.

Moodle

Es una herramienta de gestión de aprendizaje (LMS), pero más específicamente de Learning Content Management (LCMS). Es una plataforma de aprendizaje diseñada para proporcionarle a los educadores, administradores y estudiantes un sistema íntegro único, robusto y seguro para crear ambientes de aprendizaje personalizados (Moodle, 2018).

A continuación se muestra una línea del tiempo acerca de la evolución de la plataforma:

Tabla 3.1. Línea de tiempo de Moodle. (Moodle, Acerca de Moodle, 2018)

1970	1999	2002	2004
La experiencia de la infancia de Martin Dougiamas lo prepara para crear una plataforma de aprendizaje basada en internet.	En la Universidad de Curtin, Martin comienza a crear Moodle para su investigación de doctorado.	Martin lanza la primera versión 1.0 de Moodle como un sistema de código abierto para el mundo. En cuestión de meses, Moodle ya se usaba en todo el mundo.	Moodle descrito por Brent Simpson como el “Linux del mundo LMS”
2008	2015	2016	2017
Martin acepta Google-O’Reilly Open Source Award en la categoría de habilitador de educación para Moodle en su versión 1.9	Moodle se convirtió en el sistema de gestión de aprendizaje más popular en el mundo con la mayor cantidad de usuarios, más de 80 millones de usuarios en todo el mundo.	Moodle llega a más de 100 millones de usuarios registrados en todo el mundo en su versión 3.1.	Moodle cumple 15 años y anuncia una alianza de inversión con Education for The Many y comienza a acelerar sus cinco proyectos clave de crecimiento, además de abrir una oficina sede en Barcelona.
2018	2019	2020	2021
Moodle 3.5 se enfoca en el cumplimiento de GDPR (Reglamento General de Protección de Datos), usabilidad y accesibilidad mejoradas. Incluye privacidad y área de políticas en el Perfil, imágenes del curso en el tablero, íconos más claros y fuentes accesibles, sonido de grabación directa y video.	En la versión 3.8 Permite incorporar diferentes tipos de contenidos interactivos, publicar en el foro para diferentes valoraciones y calificaciones y filtrar los diferentes cursos de manera personalizada. Además, ofrece tres propuestas diferentes para la organización del calendario: mes, día o evento próximo.	Para la versión 3.9 Incorpora la acción de crear actividades H5P (Actividad Contenido Interactivo). Mejora la opción de buscar y filtrar participantes y ofrece una navegación más clara.	En la versión 3.11 se hacen correcciones y mejoras de las versiones anteriores.

Características de Moodle:

La página principal de Moodle divide sus características en tres principales ramas, a continuación, se muestran las principales características de cada una de las categorías (Moodle, 2018).

Tabla 3.2. Características principales de Moodle. (Moodle, 2018)

Características Generales	Interfaz Moderna, fácil de usar	Diseñada para ser responsiva y accesible, la interfaz de Moodle es fácil de navegar, tanto en computadoras como en dispositivos móviles.
	Tablero Personalizado	Para organizar y mostrar cursos en la forma que se desee, y poder ver en conjunto los mensajes y tareas actuales.
	Actividades y herramientas colaborativas	Permite trabajar y aprender juntos en foros, wikis, glosarios, actividades de base de datos y mucho más.
	Gestión conveniente de archivos	Con la posibilidad de arrastre y colocación de archivos desde servicios de almacenamiento en la nube, incluyendo MS OneDrive, Dropbox y Google Drive.
	Editor de texto simple e intuitivo	Con la posibilidad de dar el formato conveniente al texto y añadir multimedia e imágenes con un editor que funciona con todos los navegadores de Internet y en todos los dispositivos.
	Monitoreo del progreso	Los educadores y los educandos pueden monitorear el progreso y el grado de finalización con un conjunto de opciones para monitoreo de actividades individuales o recursos, y también a nivel del curso.
Características Administrativas	Diseño personalizable	Se personaliza fácilmente un tema de Moodle con su logo, colores o se diseña un tema propio.
	Autenticación e inscripciones masivas seguras	Más de 50 opciones para autenticación e inscripción, para añadir e inscribir usuarios a un sitio y cursos Moodle.
	Capacidad Multilingüe	Permite que los usuarios vean el contenido del curso y aprendan en su propio idioma.
	Creación masiva de cursos y fácil respaldo	Cuenta con la posibilidad de añadir cursos en lotes, respaldar y restaurar cursos grandes con facilidad.
	Gestione permisos y roles de usuario	Resuelve aspectos sobre seguridad al definir roles para especificar y gestionar el acceso de los usuarios.
	Soporta estándares abiertos	Con la posibilidad de importar y exportar fácilmente cursos IMS-LTI, SCORM y otros, hacia y desde Moodle.
	Alta interoperabilidad	Permite integrar libremente aplicaciones externas y contenidos, o crear su propio Plug-in para integraciones personalizadas.

	Gestión simple de plugins	Se puede instalar y deshabilitar Plug-ins desde adentro de una sola interfaz administrativa.
	Reportes y bitácoras detalladas	Capacidad de ver y generar reportes sobre actividad y participación a nivel de curso y de sitio.
Características para Desarrollo y Gestión del Curso	Rutas directas de aprendizaje	Permite diseñar y gestionar cursos para cumplir con diversos requisitos. Las clases pueden ser dirigidas por el instructor, auto-reguladas, mixtas o completamente en línea.
	Fomente la colaboración	Las características incluidas para la publicación colaborativa fomentan que el alumno se comprometa y realice colaboración impulsada por el contenido.
	Incruste recursos externos	Se permite enseñar con materiales e incluir tareas provenientes de otros sitios y conectarlos al libro de calificaciones en Moodle.
	Integración Multimedia	El soporte incluido en Moodle para multimedia le permite buscar fácilmente e insertar archivos de audio y video en sus cursos.
	Gestión de grupo	Permite agrupar alumnos para compartir cursos, diferenciar actividades y facilitar el trabajo en equipo.
	Flujograma de puntuación	Se puede asignar convenientemente a diferentes personas para que califiquen tareas, gestionar la moderación de calificaciones y controlar cuando se liberan las calificaciones a los alumnos individuales.
	Calificación en-línea	Da la posibilidad de revisar con facilidad y proporcionar retroalimentación en línea, al hacer anotaciones directamente dentro del navegador de Internet.
	Evaluación propia y por pares	Las actividades incluidas, tales como talleres y encuestas, estimulan a los alumnos para que vean, califiquen y evalúen el trabajo de ellos mismos y el de otros participantes del curso como un grupo.
	Resultados y rúbricas	Permite seleccionar entre métodos avanzados de calificación para personalizar el libro de calificaciones del curso y de acuerdo a sus criterios de exámenes.
	Puntuación basada en Competencias	Posibilidad de configurar competencias con planes de aprendizaje personales en cursos y actividades.
Seguridad y privacidad	Se puede enseñar y compartir dentro de un espacio privado, al que solamente pueden acceder el maestro y su grupo de alumnos.	

Actividad/Recurso “Lección”

Dentro de la plataforma de Moodle se utilizará el módulo de “Lección” ya que, según la página principal de la plataforma, la diferencia más importante entre una Lección y otros módulos de actividad disponibles en Moodle proviene de su habilidad adaptativa (Moodle, 2018).

A continuación, se muestran algunas características de las lecciones:

- Una lección está compuesta por un número de páginas y opcionalmente tablas de ramas o de contenidos.
- Cada página contiene algún tipo de contenido y, por lo general, termina con una pregunta.
- Cada página ofrece, por lo general, un número de respuestas.
- Los contenidos de la lección serán fraccionados en pequeñas partes o páginas y se mostrarán al alumno parte a parte.
- Los alumnos acceden a las diferentes páginas de la lección sólo cuando han contestado a las preguntas.
- Asociado a cada respuesta hay un salto. Este salto puede ser relativo -esta página, siguiente página- o absoluto -especificando una de las páginas de la lección o el final de la lección.
- Las respuestas correctas permiten avanzar hacia el final de la lección o rama.
- Las respuestas erróneas son penalizadas mostrando el mismo contenido otra vez o incluso alguna parte posterior adicional de la lección, o hasta el principio de la lección o rama.
- Los tipos de preguntas son: Opción Múltiple, Respuesta Corta, Verdadero/Falso, Emparejando y Numérica.
- Si el alumno no completa la lección, cuando regrese a la misma se le dará la opción de comenzar desde el principio o desde la última respuesta correcta.
- En una lección que permite Retomar, el alumno puede repetir la lección hasta conseguir la nota más alta.

PHP

Según la página principal de PHP por su acrónimo recursivo PHP: Hypertext Preprocessor (preprocesador de hipertexto) dice que es “un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML (php, 2018).

Existen tres campos principales donde se usa el lenguaje PHP (php, 2018), los cuales se muestran a continuación:

- Scripts del lado del servidor: Este es el uso principal del lenguaje PHP, en estos scripts es necesario tres cosas para que funcione, el analizador de PHP (módulo CGI-[Common Gateway Interface] o servidor), un servidor web y un navegador web.
- Scripts desde la línea de comandos: Se puede crear un script de PHP y ejecutarlo sin necesidad de un servidor o navegador, también pueden usarse para procesamiento de texto.
- Escribir aplicaciones de escritorio: Aunque no es el lenguaje más apropiado para crear aplicaciones de escritorio con Interfaz Gráfica de Usuario, pero si se desea usar PHP y utilizar sus características avanzadas puede utilizarse PHP-GTK que es una extensión de PHP, no disponible en la distribución principal. Pero se puede obtener más información de su página web.

Algunas de las características de PHP según la página principal de PHP (php, 2018) son:

- Puede utilizarse en distintos sistemas operativos como Linux, muchas variantes de Unix (HP-UX, Solaris y OpenBSD entre otros), Mac OS X, RISC OS, Microsoft Windows, entre otros.
- Admite la utilización de distintos servidores web, incluyendo Apache, IIS, entre otros.
- Permite la creación de archivos PDF, creación de imágenes e incluso películas Flash, puede generar cualquier tipo de texto como XHTML y cualquier archivo XML.
- Tiene un soporte para un amplio abanico de base de datos como MySQL, o cualquier base de datos con conexión abierta por medio de la extensión ODBC.

HTML

HTML HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto es el lenguaje con el que se define el contenido de las páginas web. Básicamente se trata de un conjunto de etiquetas que sirven para definir el texto y otros elementos que compondrán una página web, como imágenes, listas, vídeos, etc. (Alvarez, 2010).

El HTML es un lenguaje de marcación de elementos para la creación de documentos hipertexto, muy fácil de aprender, lo que permite que cualquier persona, aunque no haya programado en la vida, pueda enfrentarse a la tarea de crear una web. HTML es fácil y pronto se puede dominar el lenguaje.

Este lenguaje se escribe en un documento de texto, por eso se necesita un editor de textos para escribir una página web. Así pues, el archivo donde está contenido el código HTML es un archivo de texto, con una peculiaridad, que tiene extensión .html o .htm (es indiferente cuál utilizar). De modo que cuando programemos en HTML lo haremos con un editor de textos, lo más sencillo posible y guardaremos nuestros trabajos con extensión .html, por ejemplo mipagina.html.

A continuación, se describirán los elementos de un documento de HTML empezando por su estructura según (Mozilla, 2019):

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Mi página de prueba</title>
  </head>
  <body>
    
  </body>
</html>
```

En la estructura anterior se tiene:

<!DOCTYPE html> Es el tipo de documento. Es un preámbulo requerido. Anteriormente, cuando HTML era joven, los tipos de documento actuaban como vínculos a un conjunto de reglas que el código HTML de la página debía seguir para ser considerado bueno, lo que podía significar la verificación automática de errores y algunas otras cosas de utilidad. Sin

embargo, hoy día es simplemente un artefacto antiguo que a nadie le importa, pero que debe ser incluido para que todo funcione correctamente.

<html></html> — **el elemento <html>**. Este elemento encierra todo el contenido de la página entera y, a veces, se le conoce como el elemento raíz (root element).

<head></head> — **el elemento <head>**. Este elemento actúa como un contenedor de todo aquello que quieres incluir en la página HTML que no es contenido visible por los visitantes de la página. Incluye cosas como palabras clave (keywords), una descripción de la página que quieres que aparezca en resultados de búsquedas, código CSS para dar estilo al contenido, declaraciones del juego de caracteres, etc.

<meta charset="utf-8"> — **<meta>**. Este elemento establece el juego de caracteres que tu documento usará en utf-8, que incluye casi todos los caracteres de todos los idiomas humanos. Básicamente, puede manejar cualquier contenido de texto que puedas incluir. No hay razón para no establecerlo, y puede evitar problemas en el futuro.

<title></title> — **el elemento <title>** establece el título de tu página, que es el título que aparece en la pestaña o en la barra de título del navegador cuando la página es cargada, y se usa para describir la página cuando es añadida a los marcadores o como favorita.

<body></body> — **el elemento <body>**. Encierra todo el contenido que desees mostrar a los usuarios web que visiten tu página, ya sea texto, imágenes, videos, juegos, pistas de audio reproducibles, y demás.

CSS

Como menciona Lázaro (2010) CSS son las siglas de Cascading Style Sheets, en español Hojas de estilo en Cascada, y es una tecnología que permite crear páginas web de una manera más exacta. Gracias a las CSS se puede controlar los resultados finales de la página, pudiendo hacer muchas cosas que no se podía hacer utilizando solamente HTML, como incluir márgenes, tipos de letra, fondos, colores.

Las Hojas de Estilo en Cascada se escriben dentro del código HTML de la página web, solo en casos avanzados se pueden escribir en un archivo a parte y enlazar la página con ese archivo.

A continuación, se describirá la estructura principal de CSS como se puede ver en la figura 3.2 como menciona (Mozilla, 2016):



Figura 3.2. Estructura principal de CSS (Mozilla, 2016)

Selector: Es el elemento HTML en el que comienza la regla. Esta selecciona el(los) elemento(s) a dar estilo (en este caso, los elementos <p>). Para dar estilo a un elemento diferente, solo se cambia el selector.

Declaración: Es una sola regla como color: red; específica a cuál de las propiedades del elemento quieres dar estilo.

Propiedades: Hay diferentes maneras en las cuales puedes dar estilo a un elemento HTML. (En este caso, color es una propiedad del elemento <p>). En CSS, se selecciona qué propiedad se quiere afectar.

Valor de la propiedad: A la derecha de la propiedad, después de los dos puntos (:), se tiene el valor de la propiedad para elegir una de las muchas posibles apariencias para una propiedad determinada (hay muchos valores para color además de red).

Las otras partes importantes de la sintaxis:

- Cada una de las reglas (aparte del selector) deben estar encapsuladas entre llaves ({}).
- Dentro de cada declaración, se debe usar los dos puntos (:) para separar la propiedad de su valor.
- Dentro de cada regla, se debe usar el punto y coma (;) para separar una declaración de la siguiente.

De este modo para modificar varios valores de propiedad a la vez, solo se necesita escribirlos separados por punto y coma (;), así:

```
p {  
  color: red;  
  width: 500px;  
  border: 1px solid black;  
}
```

También se puede seleccionar varios elementos y aplicar una sola regla a todos ellos. Incluye varios selectores separados por comas (,). Por ejemplo:

```
p, li, h1 {  
  color: red;  
}
```

Existen muchos tipos diferentes de selectores. Antes, solo se han visto los selectores de elementos, los cuales seleccionan todos los elementos de un tipo dado en los documentos HTML. Sin embargo, se puede hacer selecciones más específicas que esas. Como se puede ver en la figura 3.3. están algunos de los tipos de selectores más comunes:

Nombre del selector	Qué selecciona	Ejemplo
Selector de elemento (llamado algunas veces selector de etiqueta o tipo)	Todos los elementos HTML del tipo especificado.	p Selecciona <p>
Selector de identificación (ID)	El elemento en la página con el ID especificado (en una página HTML dada, solo se permite un único elemento por ID).	#mi-id Selecciona <p id="mi-id"> y
Selector de clase	Los elementos en la página con la clase especificada (una clase puede aparecer varias veces en una página).	.mi-clase Selecciona <p class="mi-clase"> y
Selector de atributo	Los elementos en una página con el atributo especificado.	img[src] Selecciona pero no
Selector de pseudoclase	Los elementos especificados, pero solo cuando esté en el estado especificado, por ejemplo cuando el puntero esté sobre él.	a:hover Selecciona <a>, pero solo cuando el puntero esté sobre el enlace.

Figura 3.3 Tipos de selectores más comunes en CSS. (Mozilla, 2016).

CSS está basado en el modelo de cajas, Cada una de las cajas que ocupa espacio en una página web tiene propiedades como estas:

- padding (relleno), el espacio alrededor del contenido. En la figura 3.4, es el espacio alrededor del texto del párrafo.
- border (marco), es la línea que se encuentra fuera del relleno.
- margin (margen), es el espacio fuera del elemento que lo separa de los demás.

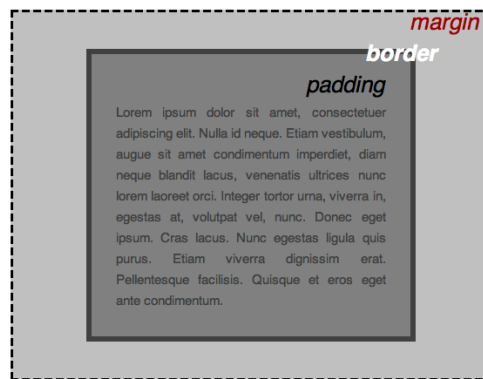


Figura 3.4. Modelo basado en cajas en CSS. (Mozilla, 2016).

JavaScript

Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript se pueden crear diferentes efectos e interactuar con los usuarios (Valdés, 2007).

Este lenguaje posee varias características, entre ellas se encuentran que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

Es necesario resaltar que hay dos tipos de JavaScript: por un lado, está el que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Pero también existe un Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript.

Entre los diferentes servicios que se encuentran realizados con Javascript en Internet se encuentran:

- Correo
- Chat
- Buscadores de Información

También podemos encontrar o crear códigos para insertarlos en las páginas como:

- Reloj
- Contadores de visitas
- Fechas
- Calculadoras
- Validadores de formularios
- Detectores de navegadores e idiomas

El código javascript se puede encontrar dentro de las etiquetas <body></body> de las páginas web. Por lo general se insertan entre: <script></script> en el mismo documento de HTML. También pueden estar ubicados en ficheros externos usando:

```
<script type="text/javascript" src="micodigo.js"></script>
```

Algunas características del lenguaje son:

Su sintaxis es similar a la usada en Java y C, al ser un lenguaje del lado del cliente este es interpretado por el navegador, no se necesita tener instalado ningún Framework.

- Variables: var = "Hola", n=103
- Condiciones: if(i<10){ ... }
- Ciclos: for(i; i<10; i++){ ... }
- Arreglos: var miArreglo = new Array("12", "77", "5")
- Funciones: Propias del lenguaje y predefinidas por los usuarios
- Comentarios para una sola línea: // Comentarios
- Comentarios para varias líneas:
/*Comentarios*/
- Permite la programación orientada a objetos: document.write("Hola");
- Las variables pueden ser definidas como: string, integer, flota, boolean simplemente utilizando "var". Podemos usar "+" para concatenar cadenas y variables.
- Es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera, Mozilla Firefox, entre otros.

Con el surgimiento de lenguajes como PHP del lado del servidor y Javascript del lado del cliente, surgió Ajax en acrónimo de (Asynchronous Javascript And XML). El mismo es una técnica para crear aplicaciones web interactivas. Este lenguaje combina varias tecnologías:

- HTML y Hojas de Estilos CSS para generar estilos.
- XMLHttpRequest es una de las funciones más importantes que incluye, que permite intercambiar datos asincrónicamente con el servidor web, puede ser mediante PHP, ASP, entre otros.

MySQL

Como menciona Robledano (2019), MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto. Desarrollado originalmente por MySQL AB, fue adquirida por Sun Microsystems en 2008 y esta su vez comprada por Oracle Corporation en 2010, la cual ya era dueña de un motor propio InnoDB para MySQL.

MySQL presenta algunas ventajas que lo hacen muy interesante para los desarrolladores. La más evidente es que trabaja con bases de datos relacionales, es decir, utiliza tablas múltiples que se interconectan entre sí para almacenar la información y organizarla correctamente.

Sus principales características son:

Arquitectura Cliente y Servidor: MySQL basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento. Cada cliente puede hacer consultas a través del sistema de registro para obtener datos, modificarlos, guardar estos cambios o establecer nuevas tablas de registros.

Compatibilidad con SQL: SQL es un lenguaje generalizado dentro de la industria. Al ser un estándar MySQL ofrece plena compatibilidad por lo que si se ha trabajado en otro motor de bases de datos no se tendrá problemas en migrar a MySQL.

Vistas: Desde la versión 5.0 de MySQL se ofrece compatibilidad para poder configurar vistas personalizadas del mismo modo que se puede hacer en otras bases de datos SQL. En bases de datos de gran tamaño las vistas se hacen un recurso imprescindible.

Procedimientos almacenados. MySQL posee la característica de no procesar las tablas directamente, sino que a través de procedimientos almacenados es posible incrementar la eficacia de nuestra implementación.

Desencadenantes. MySQL permite además poder automatizar ciertas tareas dentro de una base de datos. En el momento que se produce un evento otro es lanzado para actualizar registros u optimizar su funcionalidad.

Transacciones. Una transacción representa la actuación de diversas operaciones en la base de datos como un dispositivo. El sistema de base de registros avala que todos los procedimientos se establezcan correctamente o ninguna de ellas. En caso por ejemplo de una falla de energía, cuando el monitor falla u ocurre algún otro inconveniente, el sistema opta por preservar la integridad de la base de datos resguardando la información.

MySQL comparte las sentencias del lenguaje SQL. Lo que en la práctica se traduce en una plena compatibilidad.

- SELECT: Sirve para recuperar registros de una o más tablas.
- DISTINCT: Sirve para eliminar los duplicados de las consultas de datos.
- WHERE: Sirve para incluir las condiciones de los datos que queremos consultar.
- AND y OR: Sirve para incluir 2 o más condiciones a una consulta.
- ORDER BY: Sirve para ordenar los resultados de una consulta.
- INSERT: Sirve para insertar datos.
- UPDATE: Sirve para actualizar o modificar datos ya existentes.
- DELETE: Sirve para borrar datos.

4. DESARROLLO

Se ha elegido usar la metodología en cascada (Royce W. , 1970) para este proyecto, el cual cuenta con las siguientes fases para realización de este software: análisis, diseño, desarrollo, pruebas e implementación descritas a continuación.

4.1 Análisis.

Como parte del análisis del proyecto se utilizará la metodología de ingeniería de requisitos, curso impartido por la doctora Karla Olmos (Olmos Sánchez, 2017), donde se hace un análisis de los conceptos más significativos que integran el proyecto llamado léxico (Gladys N. Kaplan, 2009), después se realiza un modelo conceptual donde se integran los principales conceptos del léxico, casos de uso con escenarios actuales y futuros y al final una matriz de conocimiento, todo se explicará brevemente a continuación.

Como ya se mencionó parte del análisis es realizar un léxico, en el cual se van integrando palabras claves que nos ayudan a comprender de una mejor manera el proyecto, como se puede ver en la tabla 4.1 es solo un fragmento del léxico ya que mínimo se necesita alrededor de 30 conceptos.

Tabla 4.1 Fragmento de la composición del léxico.

Programación	Programación	Es el proceso de diseñar, codificar, depurar y mantener el código fuente de programas de computadora.
	Lenguaje de programación	Es un lenguaje formal diseñado para realizar procesos que puedan ser llevados a cabo por maquinas como las computadoras.
	Fundamentos de programación	Son las bases comunes a todos los programas.
	Datos	Es una representación simbólica de aun atributo o variable cuantitativa.

Estructuras sintácticas	Se entiende como el grupo de normas que marcan las secuencias correctas de los elementos propios de un lenguaje de programación.
Algoritmo	Conjunto ordenado de operaciones sistemáticas que permiten hacer un cálculo y hallar la solución de un tipo de problema.

Como siguiente etapa, se hace un modelo conceptual que sirve para describir un dominio, se expresa mediante un lenguaje de alto nivel en donde integra las principales palabras del léxico, esto ayuda a visualizar la interacción del proyecto como se muestra en la figura 4.1.

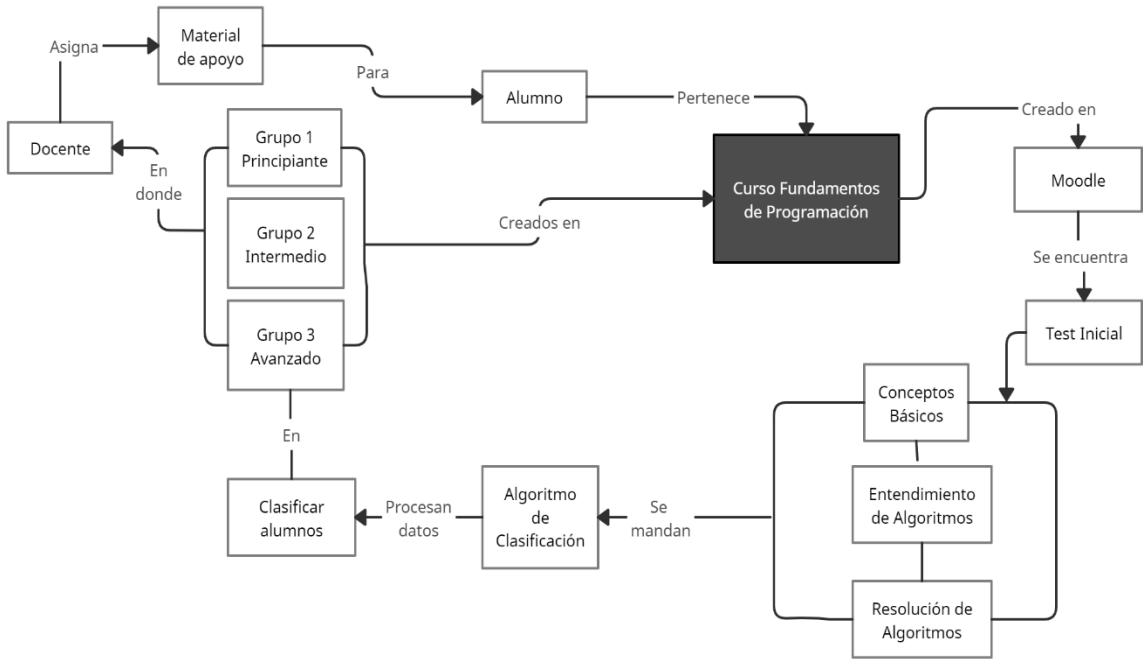


Figura 4.1. Modelo conceptual.

4.2 Diseño.

Se incluye un modelo del sistema que es más que nada los escenarios actuales o casos de uso, en donde se describe la situación o problema actual y los escenarios futuros, en donde se plantea como va a funcionar el software a desarrollar tal y como se muestra en la tabla 4.1 y tabla 4.2.

Tabla 4.1. Caso de uso en un escenario actual.

Caso de uso: Método de enseñanza actual.

Objetivo: Analizar cómo es la enseñanza en la actualidad en la materia de Fundamentos de Programación.

Episodios:

- 1.- El alumno llega al aula a tomar su clase.
- 2.- El docente da instrucciones sobre la forma de evaluar y los recursos a usar.
- 3.- El docente propone actividades para cada tema del curso.
- 4.- El docente proyecta lo que visualiza en su computadora para que los alumnos lo sigan y vean como se hace cada sentencia.
- 5.- El docente va explicando cada sección del código que va escribiendo.
- 6.- El docente aplica evaluación de la unidad.

Contexto: Aula de clases.

Recursos: Plataforma Moodle, entorno de desarrollo, computadoras.

Actores: Alumnos, docente.

Tabla 4.2. Casos de uso en escenarios futuros.

Caso de uso: Uso de la herramienta a implementar.	
Objetivo: Analizar cómo será el uso de la herramienta a implementar.	Episodios: <ol style="list-style-type: none">1.- El alumno realizará un examen para evaluar sus conocimientos en las dimensiones teóricas y prácticas.2.- El sistema con base a los resultados de la encuesta y con ayuda de algoritmos de programación clasificará al alumno en el grupo correspondiente.3.- El sistema brindará los recursos educativos al alumno según el grupo donde se encuentre.
Contexto: Aulas de clase.	
Recursos: Plataforma Moodle, entorno de desarrollo, computadoras.	
Actores: Alumnos, docente.	

Teniendo en cuenta la información anterior se plasmó en un diagrama de secuencia el funcionamiento del sistema como se ve en la figura 4.2.

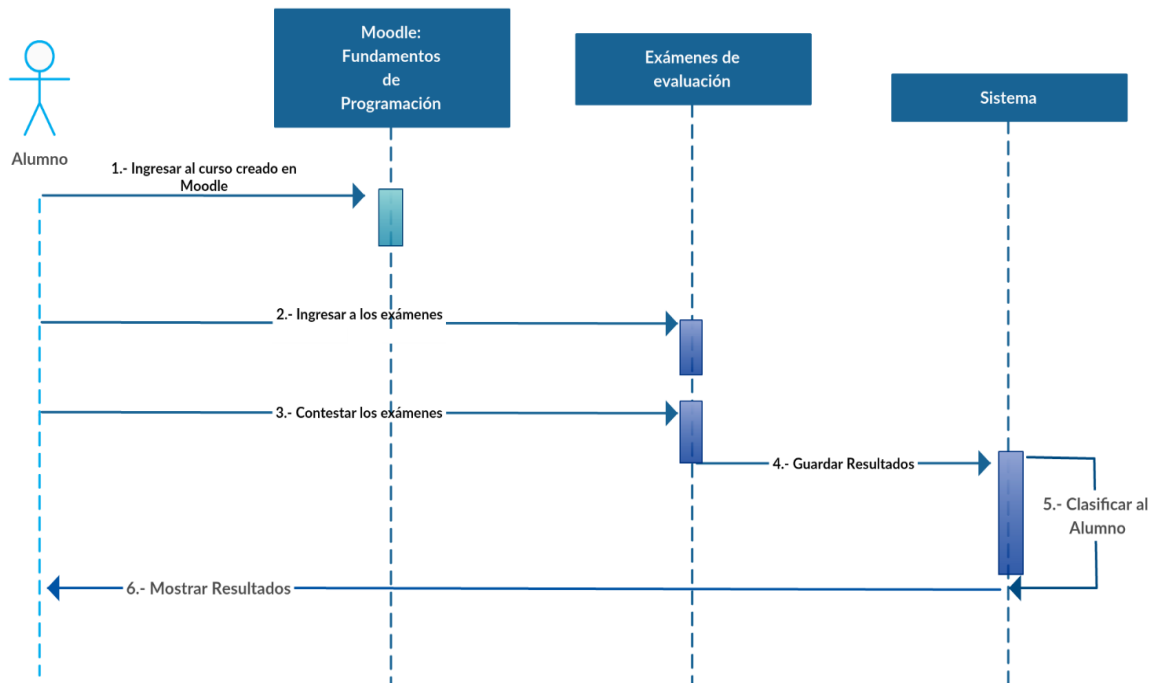


Figura 4.2. Diagrama de secuencia del sistema.

También se emplea el uso de una matriz de conocimientos en la cual se evalúa en ciertas áreas definidas a las personas involucradas en el proyecto. Se tiene un rango que va desde (-1) hasta 1, donde (-1) es que carece de conocimiento, (0) es que tiene alguna noción sobre el tema y por último el (1) que significa que cuenta con los conocimientos suficientes como se puede observar en la tabla 4.3.

Tabla 4.3. Matriz de conocimientos.

	Técnicas de programación	Fundamentos de Programación	LMS	Sistemas Adaptativos de Aprendizaje
Profesor que imparte la materia de FP	-1	1	1	0
Ingeniero de requisitos	0	1	0	0
Especialista en técnicas de programación	1	1	1	1
Director de Tesis	-1	1	1	1

4.3 Programación.

Con lo visto anteriormente el sistema funcionará de la siguiente manera: El alumno de nuevo ingreso matriculado en el curso de Fundamentos de Programación, accederá a la plataforma Moodle mediante un navegador web como se muestra en la figura 4.3.



Figura 4.3. Página principal de la plataforma Moodle.

Seleccionará el curso anteriormente mencionado como aparece en la figura 4.4.

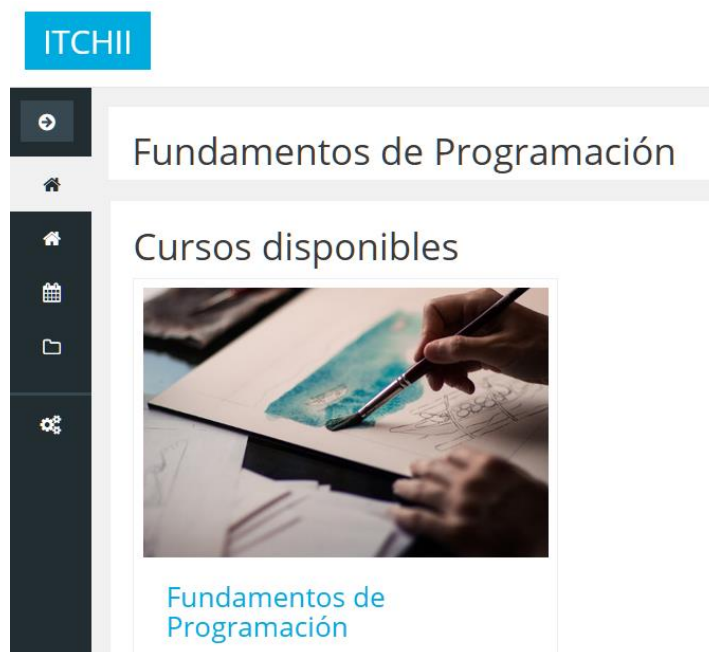


Figura 4.4. Curso disponible en la plataforma Moodle.

En primera instancia aparecerán los exámenes a contestar por el alumno que evaluarán los conocimientos como son conceptos básicos, entendimiento y resolución de algoritmos como se muestra en la figura 4.5.

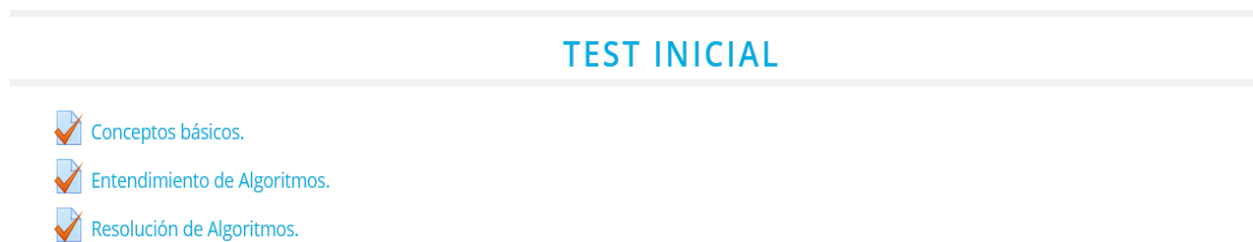


Figura 4.5. Test inicial de conocimientos.

Una vez que el estudiante conteste los exámenes se guardará la calificación en la base de datos y con ayuda de los algoritmos de programación se tomarán esos datos para que sean procesados y de esta manera se determine a cuál grupo pertenecerá al alumno, ya sea en el grupo que cuenta con conocimientos básicos, al que cuenta con conocimientos medios y por último el grupo donde cuenta con conocimientos altos tanto teóricos como capacidades de hacer algoritmos como se muestra en la figura 4.6.

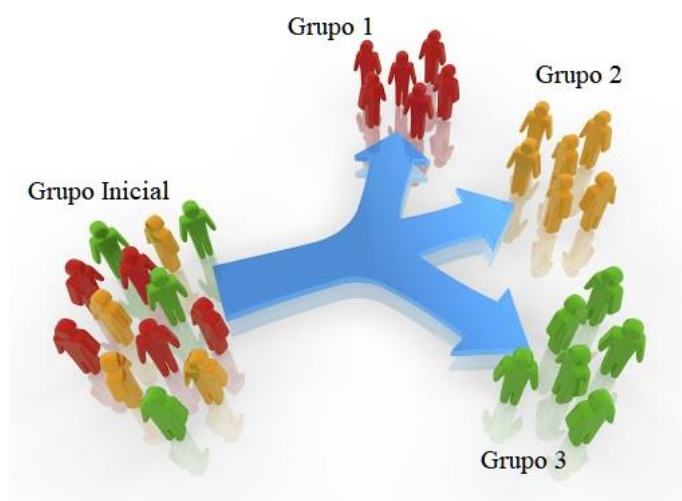


Figura 4.6. Grupos separados después de la clasificación.

En la figura 4.6 se puede ver cómo quedará la separación de los alumnos después de haber contestado los exámenes y ejecutado el algoritmo de clasificación en el grupo 1 de avanzados, grupos 2 de intermedios y grupo 3 de principiantes.

Como se mencionó anteriormente se hará uso de la base de datos por los cual se analizó la estructura de la base de datos de la plataforma Moodle la cual cuenta con alrededor de 389 tablas y se encontraron las que tienen relacionan directamente con los usuarios y los grupos como se muestra en la tabla 4.4.

Tabla 4.4. Principales tablas de Moodle usadas en el proyecto.

Nombre de la tabla:	Descripción
mdl_groups	Se encuentra información general sobre los grupos creados en Moodle, como nombre, id, etc.
mdl_groups_members	Se encuentra información relacionada con el id del grupo y el id del usuario.
mdl_user	Se encuentra información sobre los usuarios como nombre, id, password, email, entre otros registros.
mdl_course	Muestra la información de los cursos en el Moodle, como el nombre, id, etc.
mdl_role	Muestra los diferentes roles que se puede tener un usuario en la plataforma.
mdl_modules	Muestra la información de las diversas actividades con las que cuenta la plataforma para asignar a los alumnos.
mdl_quiz	Muestra la información sobre los nombres de cada examen que se encuentra en el curso.
mdl_quiz_grades	Muestra la información de las calificaciones de cada examen en el curso.

Con la información de las tablas anteriores se realizó una aplicación web para la manipulación de la base de datos de Moodle. Se creó un botón para mostrar las calificaciones de cada alumno con el nombre correspondiente a cada examen. Tal y como se muestra en la figura 4.7 y figura 4.8.

Mostrar Calificaciones del test inicial:



Figura 4.7. Mostrar calificaciones del test inicial.

Nombre del alumno	Calificaciones	Examen
Itzel Payan	10.00000	Conceptos básicos.
Itzel Payan	8.83333	Entendimiento de Algoritmos.
Itzel Payan	6.45835	Resolución de Algoritmos.
Hugo Terrazas	6.71112	Conceptos básicos.
Hugo Terrazas	7.50000	Entendimiento de Algoritmos.
Hugo Terrazas	5.20835	Resolución de Algoritmos.
Tony Stark	5.02222	Conceptos básicos.
Tony Stark	3.66668	Entendimiento de Algoritmos.
Tony Stark	4.63890	Resolución de Algoritmos.
Enrique Jaquez	7.95556	Conceptos básicos.
Enrique Jaquez	10.00000	Entendimiento de Algoritmos.
Enrique Jaquez	5.76390	Resolución de Algoritmos.

Figura 4.8. Calificaciones del alumno correspondiente a cada examen.

Como se puede observar en la figura 4.8, cada alumno aparece tres veces, esto es porque cada uno de los resultados corresponde a los exámenes realizados por el alumno en la plataforma.

Siguiendo con la manipulación de la base de datos, se creó una tabla como la anterior, pero en este caso se le agregó el Id de usuario y el Id de Examen, esto con el fin de poder relacionar todas las tablas correspondientes y así poder realizar la clasificación de los alumnos a sus respectivos grupos como se muestra en la figura 4.9.

Nombre del alumno	Calificaciones	ID de usuario	ID Examen	Nombre Examen
Itzel Payan	10.00000	3	3	Conceptos básicos.
Itzel Payan	8.83333	3	4	Entendimiento de Algoritmos.
Itzel Payan	6.45835	3	5	Resolución de Algoritmos.
Hugo Terrazas	6.71112	4	3	Conceptos básicos.
Hugo Terrazas	7.50000	4	4	Entendimiento de Algoritmos.
Hugo Terrazas	5.20835	4	5	Resolución de Algoritmos.
Tony Stark	5.02222	11	3	Conceptos básicos.
Tony Stark	3.66668	11	4	Entendimiento de Algoritmos.
Tony Stark	4.63890	11	5	Resolución de Algoritmos.
Enrique Jaquez	7.95556	6	3	Conceptos básicos.
Enrique Jaquez	10.00000	6	4	Entendimiento de Algoritmos.
Enrique Jaquez	5.76390	6	5	Resolución de Algoritmos.

Figura 4.9. Calificaciones correspondientes a cada examen con su Id respectivamente.

Como podemos ver en la figura 4.9 a la alumna Itzel Payan le corresponde el Id 3, al alumno Hugo Terrazas el Id 4, etc. En tanto a los exámenes, Conceptos básicos le corresponde el Id 3, para Entendimiento de Algoritmos corresponde el Id 4 y por último Resolución de Algoritmos el Id 5.

En la plataforma Moodle se crearon los grupos correspondientes para poder dar de alta a los alumnos que están en el curso de Fundamentos de Programación como podemos ver en la figura 4.10.

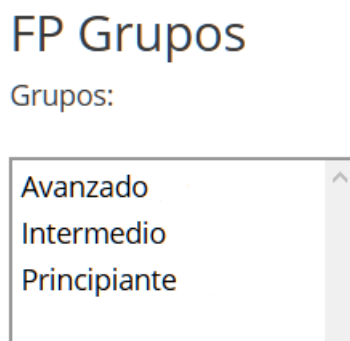


Figura 4.10. Grupos creados del curso Fundamentos de Programación.

Además de haber creado los grupos en Moodle, estos mismos se encuentran en la base de datos que servirán para extraer la información del Id del Grupo como se muestra en la figura 4.11.

id	courseid	idnumber	name
6	2		Principiante
7	2		Intermedio
8	2		Avanzado

Figura 4.11. Grupos dados de alta en la Base de datos de Moodle.

Como se puede ver en la figura 4.11, para el grupo de Principiantes le corresponde el Id 6, para el de Intermedios el Id 7 y para el Avanzado el Id 8.

Con esta información ya es posible clasificar a los alumnos mediante un algoritmo de programación y a su vez asignarlos en la plataforma Moodle al grupo correspondiente.

Dicho lo anterior se creó un botón llamado Clasificar que automáticamente consulta las calificaciones de los exámenes de cada alumno y las manda a un algoritmo para su clasificación y asignación de grupo como se muestra en la figura 4.12.

Clasificación de los Alumnos:

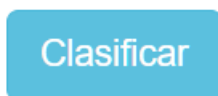


Figura 4.12. Botón para ejecutar la clasificación.

Al terminar la ejecución del algoritmo tenemos a los alumnos clasificados en sus respectivos grupos como se muestra en la figura 4.13, figura 4.14 y figura 4.15.

FP Grupos

Grupos:

Avanzado (1)	^
Intermedio (2)	
Principiante (1)	

Miembros de: Avanzado (1)

Estudiante	^
Itzel Payán	

Figura 4.13. Alumno perteneciente al Grupo Avanzado.

FP Grupos

Grupos:

Avanzado (1)	^
Intermedio (2)	
Principiante (1)	

Miembros de: Intermedio (2)

Estudiante	^
Enrique Jaquez	
Hugo Terrazas	

Figura 4.14. Alumnos pertenecientes al Grupo Intermedio.

FP Grupos

Grupos:

Avanzado (1)	^
Intermedio (2)	
Principiante (1)	

Miembros de: Principiante (1)

Estudiante	^
Antony Starkey	

Figura 4.15. Alumno perteneciente al Grupo Principiante.

Por otra parte, se crearon tres lecciones, una para cada grupo. Por cuestiones de tiempo se desarrollaron lecciones para un solo tema de la primera unidad de Diseño Algorítmico la cual se llama Conceptos Básicos. Ver figura 4.16.

Lecciones

Tópico/tema	Nombre
Tema 1	Diseño Algorítmico - Conceptos básicos (Principiantes)
Tema 1	Diseño Algorítmico - Conceptos básicos (Intermedio)
Tema 1	Diseño Algorítmico - Conceptos básicos (Avanzado)

Figura 4.16. Lecciones para cada nivel de grupo.

Cada lección cuenta con conceptos básicos del tema, así como un video explicativo al final. Ver figura 4.17, figura 4.18, figura 4.19, figura 4.20, figura 4.21 y figura 4.22.

Fundamentos de Programación

Página Principal (home)

Mis cursos

FP

Tema 1

Diseño Algorítmico - Conceptos básicos (Principiantes)

Diseño Algorítmico - Conceptos básicos (Principiantes)

Conceptos básicos.

Algoritmos.

Es un conjunto de pasos, procedimientos o acciones que nos permiten alcanzar un resultado o resolver un problema.

Etapas para la solución de un problema.

- Problema
- Análisis profundo del problema
- Construcción del algoritmo
- Verificación del algoritmo

Figura 4.17. Fragmento de la lección para el grupo de Principiantes.

Diseño Algorítmico - Conceptos básicos (Principiantes)

Video explicativo de conceptos básicos de algoritmos.

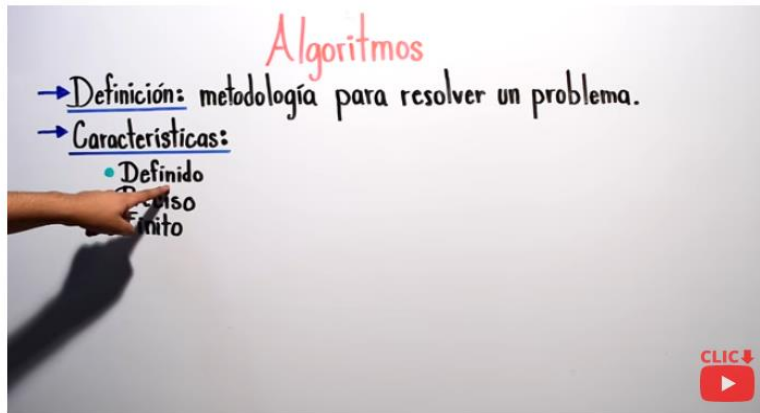


Figura 4.18. Video explicativo para el grupo de Principiantes.

Fundamentos de Programación

Página Principal (home)

Mis cursos

FP

Tema 1

Diseño Algorítmico - Conceptos básicos (Intermedio)

Diseño Algorítmico - Conceptos básicos (Intermedio)

Representación de algoritmos.

Tipos de representaciones de un algoritmo

Las principales maneras de representación de un algoritmo son a través de código fuente haciendo uso de un lenguaje de programación, a través de pseudocódigo, a través de un diagrama de flujo o a través de un sistema formal.

Código fuente

Un código fuente se define como una serie de instrucciones secuenciales, escritas en un lenguaje de programación determinado que, a través de un compilador o intérprete, éste puede ser ejecutado en una máquina.

Figura 4.19. Fragmento de lección para el grupo Intermedio.

Fundamentos de Programación

Página Principal (home)

Mis cursos

FP

Tema 1

Diseño Algorítmico - Conceptos básicos (Intermedio)

Diseño Algorítmico - Conceptos básicos (Intermedio)

Video Representacion de algoritmos.

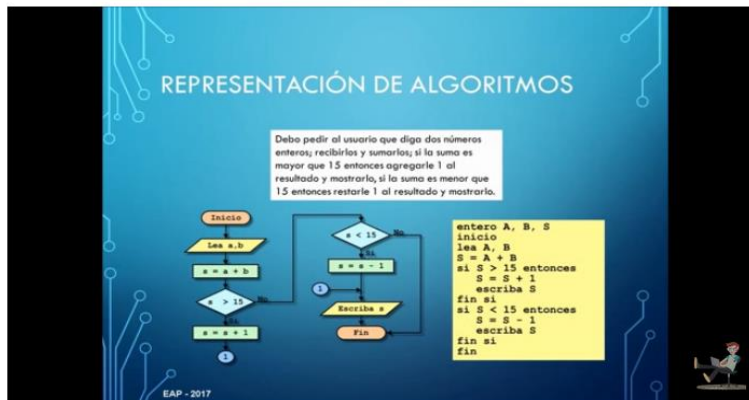


Figura 4.20. Video explicativo para el grupo de Intermedios.

Fundamentos de Programación

Página Principal (home)

Mis cursos

FP

Tema 1

Diseño Algorítmico - Conceptos básicos (Avanzado)

Diseño Algorítmico - Conceptos básicos (Avanzado)

Diseño de algoritmos.

Diseño de algoritmos:

Aunque en la solución de problemas sencillos parezca evidente la codificación en un lenguaje de programación concreto, es aconsejable realizar el diseño del algoritmo, a partir del cual se codifique el programa.

Las soluciones a problemas más complejos pueden requerir muchos más pasos. Las estrategias seguidas usualmente a la hora de encontrar algoritmos para problemas complejos son:

Figura 4.21. Fragmento de lección para el grupo Avanzado.

Fundamentos de Programación

Página Principal (home)

Mis cursos

FP

Tema 1

Diseño Algorítmico - Conceptos básicos (Avanzado)

Diseño Algorítmico - Conceptos básicos (Avanzado)

Ejemplo aplicado con Diagrama de flujo.

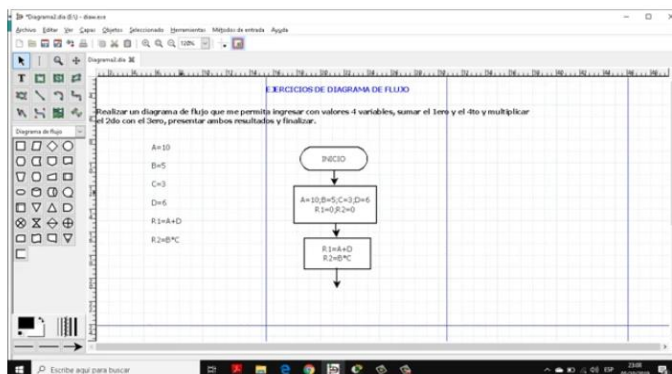


Figura 4.22. Video explicativo para el grupo Avanzado.

Se trabajó en la parte de las restricciones de los grupos, esto es que cuando el alumno sea clasificado a su correspondiente grupo dentro del curso, no podrá ver las lecciones ni los materiales de apoyo de otros grupos, tal y como se muestra en la figura 4.23.

Diseño Algorítmico.

1.1 Conceptos básicos.

1.2 Representación de algoritmos: gráfica y pseudocódigo.

1.3 Diseño de algoritmos.

1.4 Diseño de funciones

Diseño Algorítmico - Conceptos básicos (Principiantes)

Restringido No disponible, a menos que:

- Usted pertenezca a **Principiante** (de otra forma, oculto)
- Usted pertenezca a **Principiante**

Diseño Algorítmico - Conceptos básicos (Intermedio)

Restringido No disponible, a menos que:

- Usted pertenezca a **Intermedio** (de otra forma, oculto)
- Usted pertenezca a **Intermedio**

Diseño Algorítmico - Conceptos básicos (Avanzado)

Restringido No disponible, a menos que:

- Usted pertenezca a **Avanzado** (de otra forma, oculto)
- Usted pertenezca a **Avanzado**

Figura 4.23. Restricción de los grupos.

Una vez teniendo las restricciones para cada grupo, el alumno no podrá ver las lecciones de los demás grupos, solo al que le corresponde, por ejemplo, un alumno que pertenezca al grupo de avanzados no podrá ver las lecciones del grupo de intermedios ni del grupo de principiantes como se ve en la figura 4.24, figura 4.25 y figura 4.26.

The screenshot shows the ITCHII interface for 'TEMA 1'. The top navigation bar includes the ITCHII logo, a globe icon, an envelope icon, a bell icon, and the user name 'Itzel Payán' next to a profile icon. A left sidebar contains navigation icons: a right arrow, a graduation cap, a flower, a gear, a folder, and a letter 'A'. The main content area is titled 'TEMA 1' and lists the following items:

- Diseño Algorítmico.
- 1.1 [Conceptos básicos.](#)
- 1.2 Representación de algoritmos: gráfica y pseudocódigo.
- 1.3 Diseño de algoritmos.
- 1.4 Diseño de funciones
- [Diseño Algorítmico - Conceptos básicos \(Avanzado\)](#)

Figura 4.24. Restricción para alumnos del grupo Avanzado.

The screenshot shows the ITCHII interface for 'TEMA 1'. The top navigation bar includes the ITCHII logo, a globe icon, an envelope icon, a bell icon, and the user name 'Hugo Terrazas' next to a profile icon. A left sidebar contains navigation icons: a right arrow, a graduation cap, a flower, a gear, a folder, and a letter 'A'. The main content area is titled 'TEMA 1' and lists the following items:

- Diseño Algorítmico.
- 1.1 [Conceptos básicos.](#)
- 1.2 Representación de algoritmos: gráfica y pseudocódigo.
- 1.3 Diseño de algoritmos.
- 1.4 Diseño de funciones
- [Diseño Algorítmico - Conceptos básicos \(Intermedio\)](#)

Figura 4.25. Restricción para alumnos del grupo Intermedio.



TEMA 1

Diseño Algorítmico.

1.1 [Conceptos básicos.](#)

1.2 Representación de algoritmos: gráfica y pseudocódigo.

1.3 Diseño de algoritmos.

1.4 Diseño de funciones

 [Diseño Algorítmico - Conceptos básicos \(Principiantes\)](#)

Figura 4.26. Restricción para alumnos del grupo Principiantes.

4.4 Pruebas

En esta sección se explica algunas pruebas que se hicieron para comprobar algunos errores y mejoras y así al final hacer una prueba completa para validar el funcionamiento correcto del sistema.

Prueba para la obtención de información desde la base de datos.

Se creó una interfaz web en la que se pudiera recuperar la información de los alumnos inscritos en el curso de Fundamentos de Programación, así como los grupos disponibles. Ver figura 4.27 y figura 4.28.



Figura 4.27. Nombre de alumnos recuperados desde la base de datos.



Figura 4.28. Nombre de los grupos recuperados desde la base de datos.

Mediante el uso de consultas con MySQL se ha podido recuperar la información antes vista almacenada en la base de datos de Moodle de las tablas mdl_user y mdl_groups.

Prueba de datos duplicados.

En esta prueba se validó que no existieran datos duplicados en la tabla mdl_groups_members ya que a la hora de ejecutar el software insertaba de nueva cuenta los alumnos ya clasificados. Ver figura 30.

```
Validacion para ver si existen registros en la tabla de grupos
$ConsultarAlumnosGrupos = "SELECT id from mdl_groups_members
                             where groupid = $IDgrupo and
                             userid = $idEstudiante";

$ResultadoAlumnos = mysqli_query($conexion, $ConsultarAlumnosGrupos);

if (mysqli_num_rows($ResultadoAlumnos) > 0) {

    while ($fila = mysqli_fetch_array($ResultadoAlumnos)) {
        $ActualizarGrupo = "UPDATE INTO mdl_groups_members
                             (groupid, userid, timeadded, component, itemid) VALUES(" .
                             $IDgrupo . ", " . $idEstudiante . ", 1519006756, '', 0)";

        $resultadoUpdateGrupo = mysqli_query($conexion, $ActualizarGrupo);
    }
} else{

$InsertarGrupo = "INSERT INTO mdl_groups_members
                  (groupid, userid, timeadded, component, itemid) VALUES(" .
                  $IDgrupo . ", " . $idEstudiante . ", 1519006756, '', 0)";

$resultadoInsertar = mysqli_query($conexion, $InsertarGrupo);
```

Figura 4.29. Fragmento de validación de registros.

Como se puede ver en la figura 4.29, se hace una consulta a la base de datos para obtener el IDgrupo y el idEstudiante y mediante un if se validará que existan registros al comprobar que en cada fila se encuentre con datos, de lo contrario se insertará la nueva información en la tabla mdl_groups_members.

Prueba principal del sistema.

En este apartado se realizó la prueba final para ver cómo se comportaba el software y como prueba se imprimió en pantalla el nombre del alumno y al grupo correspondiente después de la ejecución del programa. Ver figura 4.30.

Clasificación = Avanzado ID Estudiante = 3 Nombre del Estudiante = Itzel Payan

ID del grupo 8

Clasificación = Intermedio ID Estudiante = 4 Nombre del Estudiante = Hugo Terrazas

ID del grupo 7

Clasificación = Intermedio ID Estudiante = 6 Nombre del Estudiante = Enrique Jaquez

ID del grupo 7

Clasificación = Principiante ID Estudiante = 11 Nombre del Estudiante = Tony Stark

Figura 4.30. Clasificación del alumno a su correspondiente grupo.

4.5 Implementación

En este proyecto se cuenta con la plataforma alojada en el servidor WampServer en su versión 3.1.0 incluyendo Apache en la versión 2.4.27 de forma local en la carpeta “www” en la raíz del disco duro, se ha instalado Moodle en su versión 3.4 para la realización de pruebas, además las herramientas web antes mencionadas han sido desarrolladas en el lenguaje de programación PHP con la versión 5.6.31 para el lado del servidor y JavaScript versión 1.7 para el lado del cliente.

En Moodle está creado el curso de Fundamentos de Programación al cual va dirigido este proyecto, en donde se ha establecido al inicio una serie de exámenes que evaluarán los conocimientos en áreas como conceptos básicos, entendimiento y elaboración sobre algoritmos como se ha mostrado con anterioridad.

El estudiante ingresará a la plataforma Moodle para acceder al curso de Fundamentos de programación, a continuación, accederá a los exámenes que aparecerán en primera instancia, al terminar los datos generados de los exámenes se recuperarán para que con la ayuda del sistema se clasifique al alumno en el grupo correspondiente según sus conocimientos previos con lo que cuente y finalmente se le proporcione los materiales necesarios para su formación académica.

5. RESULTADOS Y DISCUSIÓN

En este capítulo se describen los resultados que se han obtenido en el desarrollo de este proyecto.

En primera instancia se tiene dado de alta la plataforma Moodle en el servidor local de WampServer. Teniendo esto en cuenta, es posible ejecutar Moodle desde cualquier navegador web.

Como se puede ver en capítulos anteriores, los alumnos registrados en la plataforma pueden ingresar y darse de alta en el curso de Fundamentos de Programación y a su vez encontrar los tres exámenes de evaluación inicial.

En el primer examen llamado Conceptos Básicos se encuentran preguntas sobre los principales conceptos del tema de algoritmos donde del alumno elige entre las opciones la respuesta correcta tal y como se puede ver de ejemplo en la figura 5.1.

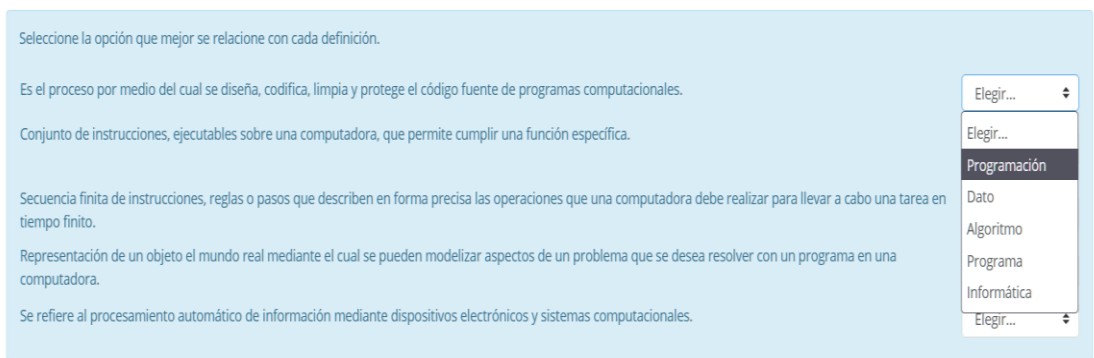


Figura 5.1. Fragmento del examen de Conceptos Básicos.

Como segundo examen se tiene el de Entendimiento de algoritmos en donde hay preguntas relacionadas sobre lo que es un algoritmo tanto de selección múltiple como arrastrar y soltar imágenes para formar secuencias como se puede ver en la figura 5.2.

Características de un algoritmo.

Seleccione una o más de una:

- A. Ser preciso
- B. Ser definido
- C. Ser finito
- D. Ser Repetitivo
- E. Todas las anteriores

Arrastre los textos a la imagen para formar la secuencia de un algoritmo.

Proceso Datos de Entrada Datos de Salida

Figura 5.2. Fragmento del examen Entendimiento de Algoritmos.

Como tercer examen se tiene el correspondiente a Resolución de algoritmos, en este apartado el alumno tiene que resolver mediante pseudocódigo problemas para obtener un resultado arrastrando y soltando los elementos hacia el área correspondiente como se puede ver en la figura 5.3.

Arrastre las palabras al papel para formar un pseudocódigo que permita resolver el área de un rectángulo:

Programa, área

Algoritmo:

Entorno: BASE, ALTURA, AREA son número enteros

leer BASE, ALTURA escribir "Introduzca la base y la altura"

escribir "El área del rectángulo es *AREA" calcular AREA = BASE * ALTURA

Finprograma

Figura 5.3. Fragmento del examen Resolución de algoritmos.

Se tienen creado tres grupos en la plataforma de Moodle, específicamente en el curso de Fundamentos de programación, los cuales son el Grupo Avanzado, Intermedio y Principiante. Cada grupo tiene material de apoyo correspondiente al nivel y restricciones en cada uno para que un alumno que no esté en el grupo no pueda ver el contenido.

Se cuenta con un algoritmo que es capaz realizar lo siguiente:

- ✓ Recuperar las calificaciones de los tres exámenes.
- ✓ Identificar la calificación correspondiente a cada examen.
- ✓ Evaluar los tres resultados de los exámenes.
- ✓ Clasificar al alumno a un grupo correspondiente en Moodle.

Todo lo anterior se realiza automáticamente al presionar un botón.

Con base en lo antes mencionado y las pruebas realizadas, se puede definir que se han logrado en un cien por ciento los objetivos establecidos en el proyecto, ya que se cuenta con un sistema que evalúa los conocimientos previos de los alumnos de nuevo ingreso, se determina a qué grupo corresponderá cada alumno y dependiendo a que grupo corresponde el alumno se le dará diferente material de apoyo académico.

6. CONCLUSIONES

Este trabajo muestra la investigación sobre la importancia de los sistemas adaptativos, como ya se sabe cada persona aprende de diferente manera y con la ayuda de estos sistemas el aprendizaje se vuelve más fácil.

Como se puede ver, se está trabajando en el desarrollo de una herramienta web que ayude a los alumnos de nuevo ingreso a tener un mejor aprendizaje en la materia de Fundamentos de Programación. Con esto lleva la integración de tecnologías como las diversas técnicas de programación y las tecnologías web para su implementación del software y así demostrar que este tipo de herramientas son de gran ayuda tanto para los alumnos como para el profesor, dejando en claro que este tipo de sistemas jamás podrán remplazar a un docente.

Con la implementación de la herramienta web se espera que los alumnos que la utilicen puedan obtener los conocimientos y las bases para su próximo curso de programación orientada a objetos y a su vez mejorar el nivel de aprovechamiento.

Como mejoras se puede aplicar lo siguiente:

- Abarcar en su totalidad todas las unidades del curso de Fundamentos de Programación.
- Extender la cantidad de material académico para cada grupo.
- La posibilidad de entrar a la plataforma y ejecución del sistema desde otro lugar que no sea la escuela.
- El uso de técnicas de programación más avanzadas para el reconocimiento de patrones de aprendizaje.

Algunas limitantes que tiene el sistema se pueden definir las siguientes:

- Las versiones del servidor y el manejador de bases de datos no se encuentran en su última versión, por lo que puede provocar incompatibilidades si se migra a versiones actuales.
- La infraestructura de redes de internet e intranet para la ejecución de la plataforma y del software.

A futuro, se espera que pueda ser utilizado para otras materias dentro del instituto y que no solo quede para el curso de Fundamentos de Programación, y de esta manera comprobar que el uso de este tipo de herramientas para la educación es de gran ayuda para los profesores y útil para los alumnos.

7. BIBLIOGRAFÍA.

- Alvarez, M. A. (1 de Enero de 2010). *desarrolloweb.com*. Obtenido de <https://desarrolloweb.com/articulos/que-es-html.html>
- Argotte, L. (2011). SI-APRENDE: An Intelligent Learning System based on SCORM Learning Objects for Training Power Systems Operators. *Springer*.
- Carlos. (s.f.). *Edmodo*. Obtenido de <https://www.edmodo.com/?language=es>
- CEDDIE*. (8 de Julio de 2015). Obtenido de <http://centrodeinnovacionytecnologia.blogspot.mx/2014/01/que-es-aprendizaje-adaptativo.html>
- Cooperativa, U. V. (2010). <http://www.ucc.edu.co>. Obtenido de http://www.ucc.edu.co:8080/learning/Documents/Manual_del_Estudiante_Blackboard_UVC.pdf
- EcuRed. (2016). <https://www.ecured.cu/Blackboard>.
- Fernández, M. O. (2018). PROMOCIÓN DE LA AUTOGESTIÓN A TRAVÉS DE OBJETOS DE APRENDIZAJE ADAPTATIVOS EN ALUMNOS DE EDUCACIÓN SUPERIOR. *Revista Electrónica de Tecnología Educativa*, 14.
- García, P. (2013). Modelo de Navegación adaptativa sobre la plataforma Moodle. *Universidad del Valle*.
- Gladys N. Kaplan, G. D. (2009). INGENIERIA DE REQUISITOS. 105.
- Guirao, M. (2011). Test de inteligencia Binet-Simon. *Psicodiagnóstico*.
- Lazaro, J. M. (1 de Enero de 2010). *desarrolloweb*. Obtenido de [desarrolloweb: https://desarrolloweb.com/articulos/26.php](https://desarrolloweb.com/articulos/26.php)
- Lerís, D. (4 de diciembre de 2015). Aprendizaje adaptativo en moodle: tres casos prácticos. *Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*, pp. 138-157.
- Llano, J. G. (13 de Noviembre de 2019). *blogs.iadb.org*. Obtenido de <https://blogs.iadb.org/conocimiento-abierto/es/aprendizaje-adaptativo-aprendizaje-abierto/>
- López, D. L. (Diciembre de 2015). Adaptive learning in moodle: three practical cases. *Evsal Revistas*.
- Lozano, M. d. (16 de septiembre de 2016). Aprendizaje Adaptativo. *Aprendizaje Adaptativo*, págs. 16-26.

- Martín, S. M. (15 de Agosto de 2012). *Columbus* . Obtenido de Columbus :
<https://grupocolumbus.blogspot.com/2012/08/la-maquina-de-skinner.html>
- Moodle. (11 de Mayo de 2018). Obtenido de Características de Moodle 3.4:
https://docs.moodle.org/all/es/Caracter%C3%ADsticas_de_Moodle_3.4
- Moodle. (11 de Mayo de 2018). *Acerca de Moodle*. Obtenido de Moodle:
https://docs.moodle.org/all/es/Acerca_de_Moodle
- Moodle. (8 de Agosto de 2018). *Actividad de lección*. Obtenido de Actividad de lección.:
https://docs.moodle.org/all/es/Actividad_de_lecci%C3%B3n
- Mozilla. (5 de Junio de 2016). Obtenido de
https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/CSS_basics
- Mozilla. (2 de Junio de 2019). *developer.mozilla.org*. Obtenido de developer.mozilla.org:
https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics
- Núñez-Torrón, A. (4 de Mayo de 2021). *Business Insider*. Obtenido de
<https://www.businessinsider.es/edmodo-como-funciona-plataforma-educativa-aula-859237>
- Olmos Sánchez, k. &. (2017). *Proceso Sistemático de Gestión del Conocimiento para el Análisis de Requisitos*. Fabro Editores. Fabro Editores.
- php. (2018). *Manual de PHP*. Obtenido de Conceptos básicos. Introducción.:
<https://www.php.net/manual/es/introduction.php>
- Robledano, A. (24 de Septiembre de 2019). *Openwebinars*. Obtenido de
<https://openwebinars.net/blog/que-es-mysql/>
- Romera, I. (16 de Junio de 2015). *UNED*. Obtenido de
http://portal.uned.es/portal/page?_pageid=93,49825468&_dad=portal&_schema=P ORTAL
- Romero, J. (2014). DISEÑO DE UN APRENDIZAJE ADAPTADO A LAS NECESIDADES DEL ALUMNO. *Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*.
- Royce, W. (1970). The Development of Large Software System.
- Royce, W. (1970). The Development of Large Software Systems.
- Sánchez, M. (2016). MeL: Modelo de adaptación dinámica del proceso de aprendizaje en eLearning. *Anales de Psicología*.

- Tabares-Morales, V. (2020). Raim: framework para la inclusión adaptativa en entornos educativos para todos. *Tecnológicas*, 23(47), 196. Obtenido de <https://doi.org/10.22430/22565337.1495>
- UIV. (21 de Marzo de 2018). *Características, Tipos y Plataformas más Utilizadas para Estudiar a Distancia*. Obtenido de Universidad Internacional de Valencia: <https://www.universidadviu.com/caracteristicas-tipos-y-plataformas-mas-utilizadas->
- Valdés, D. P. (3 de Julio de 2007). *Maestros del web*. Obtenido de <http://www.maestrosdelweb.com/que-es-javascript/>
- Won. (Septiembre de 2016). *School Illinois*. Obtenido de PLATO (Programmed Logic for Automatic Teaching Operations) system.: <http://people.ischool.illinois.edu/~chip/projects/timeline/1960won.html>

8. ANEXO A Código fuente.

Index.php

Inicio del documento en HTML donde se integran los enlaces de Scripts y recursos a utilizar en el proyecto.

```
<!DOCTYPE html>
```

```
<!--
```

To change this license header, choose License Headers in Project Properties.

To change this template file, choose Tools | Templates

and open the template in the editor.

```
-->
```

```
<html>
```

```
  <head>
```

```
    <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
```

```
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-select/1.12.4/css/bootstrap-select.min.css">
```

```
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
```

```
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
```

```
    <script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-select/1.12.4/js/bootstrap-select.min.js"></script>
```

```
    <!--<script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-select/1.12.4/js/i18n/defaults-*.min.js"></script-->
```

Ejecución del Script

```
<script>
```

```
  function ShowHint(str) {
```

```
    //Verificar si el parámetro está vacío
```

```
    if (str.length == 0) {
```

```
      document.getElementById("txtHint").innerHTML = "";
```

```
      return
```

```
    } else {
```

```
      // Crear el objeto tipo "XMLHttpRequest"
```

```
      var xmlhttp = new XMLHttpRequest();
```

```
      // Verificar si llamada AJAX fue respondida
```

```
      xmlhttp.onreadystatechange = function () {
```

```
        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
```

```
          //Acciones a realizar cuando el requerimiento es respondido con éxito
```

```
          document.getElementById("txtHint").innerHTML = xmlhttp.responseText;
```

```
        }
```

```
      }
```

```
      // Indicar el código del lado del servidor a ejecutar
```

```
      //Aquí se incluye el método "GET"
```

```
      xmlhttp.open("GET", "getit.php?texto=" + str, true);
```

```
      // Llamada al servidor con AJAX
```

```
      xmlhttp.send();
```

```
    }
```

```
  }
```

```
</script>
```

En esta parte del código se inician las pruebas para la manipulación de la base de datos y Moodle.

```
<title>Herramientas web</title>
</head>
<body>
<br>
<div class="container">
<p>Alumnos registrados</p>
Seleccionar Alumno:
<select id="" class="selectpicker" title="Seleccionar Usuario" >
```

Se realiza la conexión a la base de datos y se hace una consulta para recuperar los nombres de los alumnos en la base de datos de Moodle.

```
<?php
require("Conexion2.php");
$con = Conectar();
$sql = "SELECT username FROM mdl_user ORDER by username ASC";
$stmt = $con->prepare($sql);
//Ejecutar consulta
$result = $stmt->execute();
$rows = $stmt->fetchAll(\PDO::FETCH_OBJ);
foreach ($rows as $row) {
    ?>
    <option value="<?php print($row->username); ?>"><?php print($row->username); ?></option>
    <?php
}
?>
</select><br>
```

Se hace una consulta para recuperar los nombres de los grupos.

```
Grupo:
<select id="" class="selectpicker" title="Seleccionar Grupo">
<?php
// include("Conexion2.php");
// $con = Conectar();
    $consulta = "SELECT name FROM mdl_groups ORDER by name ASC";
    $stmte = $con->prepare($consulta);
//Ejecutar consulta
    $resultado = $stmte->execute();
    $filas = $stmte->fetchAll(\PDO::FETCH_OBJ);
    foreach ($filas as $fila) {
        ?>
        <option value="<?php print($fila->name); ?>"><?php print($fila->name); ?></option>
        <?php
    }
    ?>
</select>
<br><br>
<!--<input type="submit" value="Asignar Grupo" class="btn btn-info btn-lg" data-toggle="modal"
data-target="#myModal">-->
```

```

<!-- <div class="container">
    <h2>Modal Example</h2>
    Trigger the modal with a button
    <button type="button" class="btn btn-info btn-lg" data-toggle="modal" data-
target="#myModal">Agregar Usuario</button>-->
    <div class="modal fade" id="myModal" role="dialog">
        <div class="modal-dialog modal-sm">
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-dismiss="modal">&times;</button>
                    <h4 class="modal-title">Asignar Grupo</h4>
                </div>
                <div class="modal-body">
                    <p>Usuario Agregado</p>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-default" data-dismiss="modal">Cerrar</button>
                </div>
            </div>
        </div>
    </div>
</div>
</div>
</div>
<!--</div>-->
<hr>
<br><br>
<div class="container">
    <p>Asignar lecciones a un Grupo.</p>
    <div class="list-group">
        <a href="#" class="list-group-item">1.1 Conceptos básicos</a>
        <a href="#" class="list-group-item">1.2 Representación de algoritmos: gráfica y
pseudocódigo.</a>
        <a href="#" class="list-group-item">1.3 Diseño de algoritmos.</a>
        <a href="#" class="list-group-item">1.4 Diseño de funciones. </a>
    </div>
</div>
<div class="container">
    <form>
        <label class="checkbox-inline">
            <input type="checkbox" value="">Grupo 1
        </label>
        <label class="checkbox-inline">
            <input type="checkbox" value="">Grupo 2
        </label>
        <label class="checkbox-inline">
            <input type="checkbox" value="">Grupo 3
        </label>
        <!-- <label class="checkbox-inline">
            <input type="checkbox" value="">Grupo 4
        </label>-->
    </form>
</div>

```

```

</form>
<br>
<div class="container">

  <!-- Trigger the modal with a button -->
  <button type="button" class="btn btn-info btn-lg" data-toggle="modal" data-
target="#myModal2">Asignar lección</button>
  <!-- Modal -->
  <div class="modal fade" id="myModal2" role="dialog">
    <div class="modal-dialog modal-sm">
      <div class="modal-content">
        <div class="modal-header">
          <button type="button" class="close" data-dismiss="modal">&times;</button>
          <h4 class="modal-title">Asignar Lección</h4>
        </div>
        <div class="modal-body">
          <p>Lección Asignada.</p>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-default" data-dismiss="modal">Cerrar</button>
        </div>
      </div>
    </div>
  </div>
</div>
<br>
<hr>

```

En esta parte de código inicia la ejecución del algoritmo que se recupera las calificaciones de los tres exámenes.

```

<div class="container">
  Mostrar Calificaciones del test inicial:
  <form name="formulario" method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
    <input type="text" name="textobusqueda" id="textobusqueda" style="display: none">
    <br><br>
    <input class="btn btn-info btn-lg" type="submit" value="  Mostrar  ">
    <br><br>
  </form>
  <?php
  if (isset($_POST['textobusqueda'])) {
    // Obtener el texto a buscar
    $texto = $_POST['textobusqueda'];
    //Conexion a la base de datos
    $conexion = mysqli_connect("localhost", "root", "", "moodle");
    //Verificar el resultado de la conexion
    if (mysqli_connect_errno()) {
      echo "Error de conexion a la BD" . mysqli_connect_errno();
    }
    //Asignar codificacion para caracteres de la BD
    mysqli_set_charset($conexion, 'utf8');

    //      $consulta = "SELECT mdl_user.id,mdl_grade_grades.userid,finalgrade, username
    //                FROM mdl_user inner join mdl_grade_grades on mdl_user.id = mdl_grade_grades.userid
    //                ORDER by username DESC";
    $consulta = "SELECT mq.id, mq.name, mg.grade, mg.userid, mu.username
                FROM mdl_quiz mq, mdl_quiz_grades mg, mdl_user mu
                WHERE mq.id = mg.quiz and mg.userid = mu.id
                ORDER by mg.id";
    //      $consulta = "SELECT mdl_user.id, mdl_grade_grades.userid, username, AVG(finalgrade) as
Promedio
    //                FROM mdl_user inner join mdl_grade_grades on mdl_user.id = mdl_grade_grades.userid
    //                GROUP BY username
    //                ORDER by Promedio DESC";
    $resultado = mysqli_query($conexion, $consulta);
    // Mostrar resultado en tabla
    echo '<table class="table">';
    '<tr>';
    echo '<th>Nombre del alumno</th>';
    echo '<th>Calificaciones</th>';
    echo '<th>Examen</th>';
    '</tr>';

```

```

while ($fila = mysqli_fetch_array($resultado)) {
    echo "<tr>
    <td>" . $fila['username'] . "</td>";
    echo "<td>" . $fila['grade'] . "</td>";
    echo "<td>" . $fila['name'] . "</td>";

    echo "</tr>";
}
echo "</table>";
mysqli_close($conexion);
}
?>
<hr>
</div>
<br>
<hr>

```

Código del botón que ejecuta para mostrar calificaciones de cada alumno.

```

<!-- // Boton clasificacion -->
<div class="container">
    Clasificación de los Alumnos:
    <form name="formulario" method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
        <input type="text" name="textoClasificador" id="textoClasificador" style="display: none">
        <br><br>
        <input class="btn btn-info btn-lg" type="submit" value=" Clasificar ">
        <br><br>
    <?php
    if (isset($_POST['textoClasificador'])) {
        // Obtener el texto a buscar
        $texto = $_POST['textoClasificador'];
        //Conexion a la base de datos
        $conexion = mysqli_connect("localhost", "root", "", "moodle");
        //Verificar el resultado de la conexion
        if (mysqli_connect_errno()) {
            echo "Error de conexion a la BD" . mysqli_connect_errno();
        }
        //Asignar codificacion para caracteres de la BD
        mysqli_set_charset($conexion, 'utf8');
        $consulta2 = "SELECT mq.id, mq.name, mg.grade as grade, mg.userid as id, mu.username as
username,
                mq.id as IdExamen, mq.name as NombreExamen
                FROM mdl_quiz mq, mdl_quiz_grades mg, mdl_user mu
                where mq.id = mg.quiz and mg.userid = mu.id
                ORDER by mg.id";
        $resultado2 = mysqli_query($conexion, $consulta2);
        echo '<table class="table">';
        '</tr>';
        echo '<th>Nombre del alumno</th>';
        echo '<th>Calificaciones</th>';
        echo '<th>ID de usuario</th>';

```

```

echo '<th>ID Examen</th>';
echo '<th>Nombre Examen</th>';
'</tr>';
while ($fila = mysqli_fetch_array($resultado2, MYSQL_ASSOC)) {
    echo "<tr>
<td>" . $fila['username'] . "</td>";
    echo "<td>" . $fila['grade'] . "</td>";
    echo "<td>" . $fila['id'] . "</td>";
    echo "<td>" . $fila['IdExamen'] . "</td>";
    echo "<td>" . $fila['NombreExamen'] . "</td>";
    echo "</tr>";
}
echo "</table>";

```

En esta sección inicia la codificación del algoritmo que clasifica a los alumnos en sus respectivos grupos.

```

// Obtener datos de cada usuario
$consulta3 = "SELECT mu.id as ID, mu.username as Nombre
FROM mdl_user mu
ORDER by mu.id";
$resultado3 = mysqli_query($conexion, $consulta3);
echo '<table class="table">';
'</tr>';
echo '<th>ID del alumno</th>';
echo '<th>Nombre Alumno</th>';
'</tr>';
while ($fila2 = mysqli_fetch_array($resultado3, MYSQL_ASSOC)) {
    echo "<tr>
<td>" . $fila2['ID'] . "</td>";
    echo "<td>" . $fila2['Nombre'] . "</td>";
}
// Obtener las tres calificaciones de cada estudiante
$idEstudiante = $fila2['ID'];
// Volver a consultar calificaciones de estudiantes
$consulta4 = "SELECT mq.id, mq.name, mg.grade as grade, mg.userid as id, mu.username as
username,
mq.id as IdExamen, mq.name as NombreExamen
FROM mdl_quiz mq, mdl_quiz_grades mg, mdl_user mu
where mq.id = mg.quiz and mg.userid = mu.id and mu.id = $idEstudiante
ORDER by mg.id";

$resultado4 = mysqli_query($conexion, $consulta4);
$Examen1 = 0;
$Examen2 = 0;
$Examen3 = 0;
$Contador = 0;
$NombreEstudiante = "";
while ($fila4 = mysqli_fetch_array($resultado4, MYSQL_ASSOC)) {
// Leer las tres calificaciones de cada estudiante
if ($fila4["NombreExamen"] == 'Conceptos básicos.') {

```



```

        $Examen1 = $fila4["grade"];
        $Contador++;
    }
    if ($fila4["NombreExamen"] == 'Entendimiento de Algoritmos.') {
        $Examen2 = $fila4["grade"];
        $Contador++;
    }
    if ($fila4["NombreExamen"] == 'Resolución de Algoritmos.') {
        $Examen3 = $fila4["grade"];
        $Contador++;
    }
    $NombreEstudiante = $fila4['username'];
}

// Obtener clasificacion
$clasificacion = "Sin Grupo";
//Obtener promedio aritmetico
$c1 = $Examen1;
$c2 = $Examen2;
$c3 = $Examen3;
$promedio = ($c1 + $c2 + $c3) / 3;
if ($promedio >= 8.0) {
    $clasificacion = "Avanzado";
} elseif ($c3 >= 9.0) {
    $clasificacion = "Avanzado";
} elseif ($promedio >= 5.0 && $promedio < 8.0) {
    $clasificacion = "Intermedio";
} elseif ($c3 >= 8.0 && $c3 < 9.0) {
    $clasificacion = "Intermedio";
} elseif ($promedio >= 0 && $promedio < 5.0) {
    $clasificacion = "Principiante";
}
//Mostrar clasificacion
echo ' Clasificación = ' . $clasificacion;
echo ' ID Estudiante = ' . $idEstudiante;
echo ' Nombre del Estudiante = ' . $NombreEstudiante;
echo '<br><br>';
if ($Contador == 3) {
// Obtener ID del grupo en que fue clasificado el estudiante.
    $Consulta5 = "SELECT id FROM mdl_groups
        WHERE name like '%$clasificacion%'";
    $resultado5 = mysqli_query($conexion, $Consulta5);
// Leer el ID del grupo
    if ($resultado5) {
        $NumFilas = mysqli_num_rows($resultado5);
    }
// Validar resultado de la consulta.
    if ($resultado5 && $NumFilas == 1) {
        $Fila5 = mysqli_fetch_array($resultado5, MYSQL_ASSOC);
        $IDgrupo = $Fila5['id'];
        echo 'ID del grupo ' . $IDgrupo;

```

```

        echo '<br><br>';
//      Agregar usuario al grupo que corresponde según su avance.
    } else {
        echo 'No existe este grupo.';
        echo '<br><br>';
    }

//      Validación para ver si existen registros en la tabla de grupos
$ConsultarAlumnosGrupos = "SELECT id from mdl_groups_members
        where groupid = $IDgrupo and
        userid = $idEstudiante";
$ResultadoAlumnos = mysqli_query($conexion, $ConsultarAlumnosGrupos);
if (mysqli_num_rows($ResultadoAlumnos) > 0) {
    while ($fila = mysqli_fetch_array($ResultadoAlumnos)) {
        $ActualizarGrupo = "UPDATE INTO mdl_groups_members
            (groupid, userid, timeadded, component, itemid) VALUES(" .
            $IDgrupo . ", " . $idEstudiante . ", 1519006756, ", 0)";

        $resultadoUpdateGrupo = mysqli_query($conexion, $ActualizarGrupo);
    }
} else{

```

En esta sección se insertan los alumnos al grupo correspondiente una vez validado que no exista registro del alumno.

```

        $InsertarGrupo = "INSERT INTO mdl_groups_members
            (groupid, userid, timeadded, component, itemid) VALUES(" .
            $IDgrupo . ", " . $idEstudiante . ", 1519006756, ", 0)";
        $resultadoInsertar = mysqli_query($conexion, $InsertarGrupo);
        echo "</tr>";
//      Cierre del if if($Contador == 3)
    }
}
}
echo "</table>";
mysqli_close($conexion);
}
// }
?>
</div>
</body>
</html>

```