



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica  
Universitat Politècnica de València

# Aplicación web para la enseñanza y aprendizaje de matemáticas

Trabajo Fin de Grado

**Grado en Ingeniería Informática**

**Autor:** Miguel Hernán González

**Tutor:** María José Labrador Piquer

Curso 2016 / 2017



*“Si avanzo, seguidme; si me detengo, empujadme; si retrocedo, matadme...Che Guevara.”*

*Gracias María.*



## Resumen

---

Mediante este proyecto planteamos la resolución de un trabajo de ingeniería cuyo objetivo es desarrollar una aplicación web, que utilizarán los estudiantes de Educación Secundaria Obligatoria (ESO), para ejercitar matemáticas y realizar las tareas obligatorias solicitadas por los profesores. A su vez la aplicación ofrece una modalidad para los docentes que pueden visualizar quién realizó las tareas, cuál es su calificación de cada tema y por consiguiente qué temas deberían reforzarse en clase.

La aplicación utiliza algoritmos inteligentes, que permiten infinitas versiones de un mismo ejercicio, por lo que cada alumno realiza uno diferente, y a su vez les permite practicar un mismo tema todas las veces que deseen siempre con distintas variables. Los ejercicios se autocorrigien y se van adaptando al nivel del estudiante, es decir, que si el alumno está cometiendo muchos errores, recibirá unos más fáciles hasta que afiance los conocimientos, y a medida que vaya acertando en las respuestas se irán dificultando.

**Palabras clave:** Aplicación web, enseñanza, aprendizaje, matemáticas, automatización.

## Abstract

---

Through this project we propose the resolution of an engineering work whose objective is to develop a web application, which students of compulsory secondary education (ESO) will use to exercise mathematics and perform the obligatory tasks requested by teachers. In turn, the application offers a modality for teachers, who can visualize who performed the tasks, what is their qualification for each topic and therefore what subjects should be reinforced in class.

The application uses intelligent algorithms, which allow infinite versions of the same exercise, so that each student performs a different one, and in turn allows them to practice the same theme every time they always want different variables. The exercises are self-correcting and adapted to the student's level, meaning that if the student is making many mistakes, he will receive some easier ones until he consolidates his knowledge, and as he gets the answers right, they will become more difficult.

**Keywords:** Web application, teaching, learning, mathematics, automation.

## Resum

---

Mitjançant aquest projecte plantegem la resolució d'un treball d'enginyeria amb l'objectiu de desenvolupar una aplicació web, que utilitzaran els estudiants d'educació secundària obligatòria (ESO), per exercitar matemàtiques i realitzar les tasques obligatòries sol·licitades pels professors. Al seu torn l'aplicació ofereix una modalitat per als docents, que poden visualitzar qui va realitzar les tasques, quina és la seva qualificació de cada tema i per tant quins temes haurien de reforçar-se en classe.

L'aplicació utilitza algoritmes intel·ligents, que permeten infinites versions d'un mateix exercici, de manera que cada alumne realitza un de diferent, i al seu torn els permet practicar un mateix tema totes les vegades que vulguin sempre amb diferents variables. Els exercicis es autocorrigen i es van adaptant al nivell de l'estudiant, és a dir que si l'alumne està cometent molts errors, rebrà uns més fàcils fins que consolidi els coneixements, i a mesura que vagi encertant en les respostes s'aniran dificultant.

**Paraules clau:** Aplicació web, ensenyament, aprenentatge, matemàtiques, automatització.

# Índice general

---

<b>CAPÍTULO 1</b> .....	<b>1</b>
Introducción.....	1
1.1 Motivación .....	1
1.2 Objetivos .....	2
1.3 Estructura de la memoria .....	2
<b>CAPÍTULO 2</b> .....	<b>3</b>
Estado de la cuestión .....	3
2.1 Plataformas educativas.....	3
2.2 Aplicaciones existentes en matemáticas.....	6
2.3 Tecnologías utilizadas para el desarrollo de la aplicación .....	11
2.3.1 Lenguaje de programación: Ruby.....	11
2.3.2 Framework Ruby on Rails.....	12
2.3.3 Base de datos: Sqlite3 .....	14
2.3.4 Editor de código utilizado: Sublime Text .....	15
<b>CAPÍTULO 3</b> .....	<b>17</b>
Análisis.....	17
3.1 Análisis de requisitos .....	17
3.2 Casos de uso .....	19
3.3 Diagrama de clases .....	21
<b>CAPÍTULO 4</b> .....	<b>23</b>
Diseño .....	23
4.1 Contexto matemático .....	23
4.2 Estructura de la aplicación .....	24
4.3 Características intrínsecas del diseño.....	24
4.4 Prototipo de la interfaz gráfica de usuario .....	25

<b>CAPÍTULO 5</b> .....	<b>27</b>
Implementación .....	<b>27</b>
5.1 Aprendizaje adaptativo.....	27
5.2 Posibles respuestas .....	28
5.3 Explicación de respuesta incorrecta.....	30
5.4 Panel profesor.....	31
5.5 Diseño gráfico de la aplicación.....	32
<b>CAPÍTULO 6</b> .....	<b>33</b>
Conclusiones.....	<b>33</b>
6.1 Dificultades encontradas.....	33
6.2 Consideraciones finales .....	34
6.3 Líneas futuras .....	34
<b>Bibliografía</b> .....	<b>35</b>
<b>APÉNDICE 1</b> .....	<b>37</b>
Imágenes de la primera versión de Talento .....	<b>37</b>
<b>APÉNDICE 2</b> .....	<b>41</b>
Objetivos didácticos específicos de 1º de E.S.O.....	<b>41</b>



# Índice de figuras

---

Ilustración 1. Logo Moodle .....	4
Ilustración 2. Logo Edmodo .....	5
Ilustración 3. Logo de ZonaClic .....	7
Ilustración 4. Funcionamiento de la aplicación <i>ZonaClic</i> .....	8
Ilustración 5. Logo Descartes .....	8
Ilustración 6. Funcionamiento de la aplicación Descartes .....	9
Ilustración 7. Logo de Educarex .....	9
Ilustración 8. Funcionamiento de la aplicación Educarex .....	10
Ilustración 9. Logo Ruby .....	11
Ilustración 10. Logo Ruby on Rails .....	12
Ilustración 11. Ciclo de vida del MVC .....	13
Ilustración 12. Logo SQLite .....	14
Ilustración 13. Logo Sublime Text .....	15
Ilustración 14. Casos de uso del administrador .....	19
Ilustración 15. Casos de uso del Alumno .....	20
Ilustración 16. Casos de uso del profesor .....	20
Ilustración 17. Diagrama de clases .....	21
Ilustración 18. Prototipo de la interfaz gráfica de usuario .....	26
Ilustración 19. Cambio de nivel de ejercicio .....	28
Ilustración 20. Posibles respuestas .....	29
Ilustración 21. Explicación de respuesta incorrecta .....	31
Ilustración 22. Panel profesor .....	31
Ilustración 23. Pantalla de inicio .....	37
Ilustración 24. Nuestra aplicación .....	37
Ilustración 25. Contacto .....	38
Ilustración 26. Inicio estudiantes .....	38
Ilustración 27. Alumno, tareas .....	39
Ilustración 28. Alumno, estadísticas .....	39
Ilustración 29. Profesores, panel administración .....	40
Ilustración 30. Estadísticas de un alumno en concreto .....	40



---

# CAPÍTULO 1

---

## Introducción

En este primer capítulo exponemos las razones que nos han conducido a la elaboración de este trabajo, así como los objetivos que pretendemos alcanzar, y una pequeña descripción de la estructura del trabajo realizado.

### 1.1 Motivación

---

El motivo de este proyecto viene dado por la vocación y la satisfacción que produce en su autor la didáctica y su deseo de llegar a ser profesor un día y poder mejorar el sistema de enseñanza de las matemáticas. En la actualidad disponemos de potentes dispositivos (móviles, tabletas, ordenadores, etc.) que llevamos con nosotros casi en cualquier momento, por lo que el uso de estos dispositivos para acceder al sistema y poder ejercitar o realizar tareas en el momento y lugar que el estudiante desee será la clave del éxito de esta aplicación denominada "Talento".

Tras realizar una búsqueda de aplicaciones con un propósito similar, encontramos que no hay ninguna que se adapte a nuestras expectativas, puesto que deseamos minimizar el trabajo de los docentes y que ya no tengan que pasar horas corrigiendo tareas en casa, y que cada alumno reciba un ejercicio con variables diferentes cuando deben hacer una tarea.

Los alumnos deben poder practicar todas las veces que quieran sin que se repitan los ejercicios, los manuales tienen los problemas contados y a veces no son suficientes para que los estudiantes incorporen el conocimiento. Todos los alumnos no son iguales, la enseñanza debe adaptarse a cada uno y los docentes no son máquinas.

Esta aplicación va dirigida a los colegios, para que sea utilizada como herramienta de ayuda para los profesores y alumnos de matemáticas.

## 1.2 Objetivos

---

El objetivo principal de este proyecto es crear una primera versión de una aplicación web para la enseñanza y aprendizaje de las matemáticas.

Características de la aplicación:

- Generación automática de ejercicios para practicar con infinitas variables.
- Corrección automática de ejercicios y explicación en caso de error.
- Informes globales e individuales para que los profesores sepan quien hizo la tarea y obtención de notas de cada alumno.
- Adaptación de los ejercicios al nivel de cada alumno.

## 1.3 Estructura de la memoria

---

El presente trabajo se compone de seis capítulos, y en esta sección, nos disponemos a realizar una breve descripción del trabajo realizado en cada uno de ellos. Dada nuestra inexperiencia en este tipo de trabajos, tomamos como referencia otros proyectos del área de la informática [1]–[4].

En el primer capítulo se exponen las razones que nos han conducido a la elaboración de este trabajo, así como los objetivos esenciales a alcanzar.

En el segundo, abordamos el estado de la cuestión, que está destinado a analizar las aplicaciones existentes y la tecnología utilizada para resolver los objetivos planteados.

En el tercero, se analiza qué es lo que se necesita para cumplir con los objetivos.

El cuarto se ocupa de realizar un estudio del contexto matemático y se diseña una estructura de archivos que soporte el proyecto.

En el quinto se detalla cómo se implementaron las funciones más destacadas. Y se cierra la estructura con las conclusiones donde se recogen las principales obtenidas, y las futuras líneas con las que podría seguir el presente proyecto.

---

# CAPÍTULO 2

---

## Estado de la cuestión

En la primera parte de este capítulo analizamos las aplicaciones similares existentes más utilizadas por los docentes en la actualidad y en la segunda se analiza la tecnología utilizada para resolver los objetivos planteados en el capítulo 1.

### 2.1 Plataformas educativas

---

Sin lugar a dudas, muchas son las ventajas y beneficios que ofrecen las Tecnologías de Información y Comunicación (TIC) como medios para la educación a distancia:

- Independencia en tiempo y en espacio: aprender en cualquier sitio y momento.
- Acceso de todos a la educación.
- Acceso a través de Internet a recursos y servicios educativos en permanente crecimiento.
- Formación bajo demanda.
- Enseñanza / aprendizaje a distancia.

Sin embargo, las TIC si se utilizan correctamente, pueden generar grandes beneficios también a la educación presencial:

- Elección del estilo de enseñanza.
- Seguimiento y registro individual de los procesos educativos.
- Autoevaluación y monitorización del rendimiento del alumno.
- Comunicación interactiva entre los agentes que participan o influyen en los procesos educativos.

Actualmente existen muchas plataformas gestoras de aprendizaje basadas en software libre: *Manhatan*, *Moodle*, *Edmodo* entre otras. La mayoría se encuentran disponibles de forma gratuita en Internet.

## Moodle



**Ilustración 1. Logo Moodle**

Es una de las plataformas educativas de software libre más populares y su comunidad de usuarios y desarrolladores es muy numerosa. Es un sistema muy flexible que funciona prácticamente en cualquier plataforma y es muy fácil de administrar y operar.

Una de las características de Moodle es la de transmitir información. El profesor se constituye prácticamente en emisor y los alumnos en receptores. Lo más usual es que lo que se transmite esté constituido por algún tipo de texto más o menos estructurado, con apoyo de imágenes, esquemas, etc. A estos elementos Moodle los llama recursos y son un enlace a los recursos materiales construidos por el docente, como documentos de texto, presentaciones con diapositivas, archivos de imagen, de audio y video, etc.

Otra característica son los recursos interactivos, los cuales se centran más en el alumno, quien tiene el control de navegación sobre los contenidos. Por ejemplo las lecciones, que se componen de una serie de páginas o textos que el alumno ha de recorrer, normalmente se configura de manera que al final de cada página se plantee una pregunta para comprobar que el alumno lo ha leído y/o comprendido y según la opción que escoja, el propio recurso le permitirá avanzar en el recorrido o le obligará a retroceder. Otro ejemplo son los cuestionarios que permiten construir listas de preguntas con diferentes tipos de respuesta, con la ventaja de la retroalimentación inmediata al alumno. Finalmente cabe mencionar a los recursos colaborativos, como los foros, o los talleres de trabajo en grupo.

Moodle promueve una pedagogía constructivista social (colaboración, actividades, reflexión crítica, etc.), y es una herramienta apropiada para el aprendizaje en línea y para complementar el aprendizaje presencial. Tiene una interfaz de navegación sencilla, ligera y eficiente, y un profesor tiene control total sobre todas las opciones de un curso, por ejemplo puede definir sus propias escalas para calificar o puede elegir entre formatos de curso semanal, por temas, pestañas, menú, social, etcétera [5].

## Edmodo



**Ilustración 2. Logo Edmodo**

Se trata de una red social, de carácter gratuito, dirigida especialmente al mundo de la educación, tanto a profesorado como a alumnado.

Tiene las aplicaciones habituales de las redes sociales profesionales, que se fundamentan en el hecho de poder compartir información y permitir hacer grupos diferentes, con lo que se pueden agrupar los alumnos en función del grado o rango, así como de un área o tema. El docente asigna un código a cada grupo y los estudiantes podrán acceder a los mismos a través de ese número con la finalidad de debatir, compartir información, asignarse tareas o enviar mensajes, entre otras aplicaciones.

En cuanto al trabajo directo con el alumnado, otra de las aplicaciones que incluye es la creación de asignaciones, es decir, proponer tareas al alumnado que, posteriormente, el docente puede calificar también a través de la red. Por otro lado, puede incluir en el calendario fechas claves, por ejemplo, para la entrega de una actividad.

Asimismo, ofrece a los docentes la opción de conectarse con los padres y madres del alumnado, con la finalidad de que sigan el trabajo y la evolución de sus hijos. De esto modo, los profesores pueden intercambiar mensajes con ellos, proporcionarles acceso directo a las notas y tareas, enviarles notificaciones acerca de las próximas fechas de vencimiento y tareas perdidas o alertarlos ante próximos eventos y actividades escolares.

Edmodo es una plataforma educativa que funciona igual que una red social, al estilo de Facebook, Tuenti o Twitter. Tiene todas la ventajas de este tipo de webs, pero sin los peligros que las redes sociales abiertas tienen, ya que se trata de crear un grupo cerrado entre el alumnado y el profesor, para compartir mensajes, enlaces, documentos, eventos, etc.

Es una red que permite la comunicación entre estudiantes, docentes y padres de familia, de forma diferente y efectiva [6].

## Reflexiones

Actualmente las universidades de España tienen su propia plataforma educativa. Algunos ejemplos son la “aLF” de la Universidad Nacional de Educación a Distancia, o “Poliformat” de la Universidad Politécnica de Valencia. En cambio los centros escolares no disponen de estos recursos, por lo que terminan acudiendo en consecuencia a los mencionados Moodle o Edmodo.

Cabe destacar que en ambos casos los docentes lo utilizan para insertar información o ejercicios, por lo que requieren un trabajo constante del profesor. Ambas son herramientas genéricas, y pueden utilizarse para cualquier tipo de asignaturas. Y es en este punto donde detectamos la posibilidad de mejoras, desarrollando una aplicación específica para matemáticas, con la posibilidad de tener todos los ejercicios del curso cargados en el sistema, y que los mismos se adapten al nivel educativo de cada alumno, lo cual para las plataformas mencionadas se hace imposible por su carácter genérico.

## 2.2 Aplicaciones existentes en matemáticas

---

En el mercado podemos encontrar muchas aplicaciones destinadas a la enseñanza de matemáticas pero ninguna de ellas nos ofrece la posibilidad de seguir la evaluación constante de toda la clase, con las notas de cada alumno y las estadísticas grupales. A su vez ninguna permite a los alumnos practicar con ejercicios que se adapten a su nivel, lo cual lo consideramos clave, dado que no todos los estudiantes aprenden a la misma velocidad.

## Criterios de elección de las aplicaciones

---

Hemos buscado y analizado aplicaciones similares y los criterios seguidos para su elección son los siguientes:

- Aplicación gratuita y disponible en internet.
- Que no requiera ningún tipo de instalación.
- Habitualmente utilizadas por docentes (se realizó una entrevista a la profesora de matemáticas Teresa Aliacar quien nos informó de cuáles son las aplicaciones que están utilizando en los institutos hoy en día).



## Aplicaciones elegidas para su estudio

---

De acuerdo con los criterios mencionados en el epígrafe anterior se han encontrado un buen número de herramientas con la misma funcionalidad principal, por lo que nos centramos en las siguientes:

### ZonaClic

---



**Ilustración 3. Logo de ZonaClic**

*ZonaClic* está formado por un conjunto de aplicaciones de software libre que permiten crear diversos tipos de actividades educativas multimedia. La ZonaClic es un servicio del Departamento de Educación de la Generalitat de Cataluña creado con el objetivo de dar difusión y apoyo al uso de estos recursos. Abarca las áreas de matemáticas, lenguas, música y muchas otras áreas.

El entorno gráfico es poco amigable y los ejercicios consisten en completar el espacio vacío con el dato correcto.

Utiliza una distinción por colores para detectar dónde se ha cometido un error.

**casos:**

a)  $(2 + 4) + 7 = 2 + (4 + 7)$   
 $(2 + 4) + 7 = 2 + 4 = 3$   
 $2 + (4 + 7) = 2 + 11 = 13$   
Por tanto:  $(2 + 4) + 7 = 2 + (4 + 7)$

b)  $(3 \cdot 5) \cdot 8 = 3 \cdot (5 \cdot 8)$   
 $(3 \cdot 5) \cdot 8 = 15 \cdot 8 = 120$   
 $3 \cdot (5 \cdot 8) = 3 \cdot 40 = 120$   
Por tanto:  $(3 \cdot 5) \cdot 8 = 3 \cdot (5 \cdot 8)$

c)  $5 + (4 + 3) = (5 + 4) + 3$   
 $5 + (4 + 3) = 12 = 12$   
 $(5 + 4) + 3 = 12 = 13$   
Así que:  $5+4+3 = 12$

d)  $7 \cdot (2 \cdot 3) = (7 \cdot 2) \cdot 3$   
 $7 \cdot (2 \cdot 3) = 7 \cdot 6 = 42$   
 $(7 \cdot 2) \cdot 3 = 14 \cdot 3 = 42$   
Por tanto:

**Ilustración 4. Funcionamiento de la aplicación ZonaClic**

Con respecto a nuestra área de interés (matemáticas), presenta una gran cantidad de ejercicios de práctica, los cuales se repiten si volvemos a hacerlos, es decir que los alumnos pueden memorizar las respuestas y al volver a realizarlos ya no aportan nada al estudiante. A su vez no se adapta al nivel del alumno cuando está practicando.

## Descartes

---



**Ilustración 5. Logo Descartes**

"Descartes" es un proyecto educativo que reúne gran cantidad y variedad de materiales didácticos y en el que participa un buen número de profesores.

Estos materiales, destinados al aprendizaje de las matemáticas de la enseñanza secundaria, cubren prácticamente la totalidad de los contenidos de los currículos de los niveles de Primaria, ESO y Bachillerato.

El alumnado, además de disponer de los contenidos necesarios, puede adquirir y relacionar conceptos, aventurar hipótesis y comprobar su validez,

hacer deducciones, establecer propiedades y teoremas, plantear y resolver problemas y, en general, realizar todas las actividades propias del aprendizaje de las matemáticas.

Es una aplicación muy completa pero que no contempla una visión para que los docentes puedan seguir las evaluaciones de los alumnos.

## Tema 1

### 1 Los números naturales

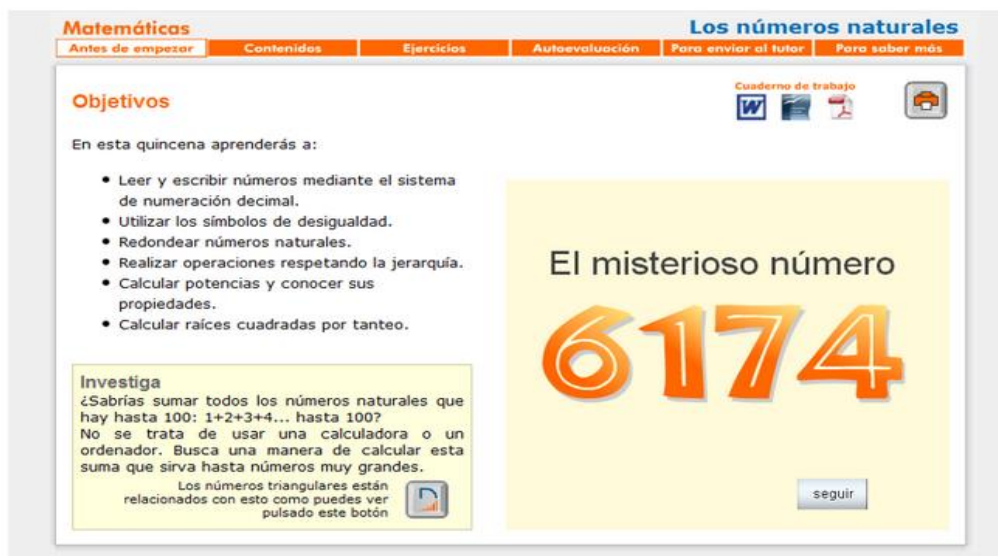


Ilustración 6. Funcionamiento de la aplicación Descartes

## Educarex



Ilustración 7. Logo de Educarex

Con una estética divertida y atractiva, la Consejería de Educación de la Junta de Extremadura ofrece una completa página en la que se pueden encontrar múltiples recursos divididos por ciclos educativos: Educación Infantil, Primaria, Secundaria y Educación Especial.

Se trata de contenidos digitales educativos curriculares y descargables que cualquier docente puede emplear en sus clases como lo crea conveniente.

A partir de Educación Primaria, cada material está estructurado por materias o temas: en el caso de 1º, se puede elegir entre Conocimiento del medio, Inglés, Lengua y literatura, Matemáticas con multitud de ejercicios: sumas y restas, geometría, multiplicaciones, etcétera.

Es bastante intuitiva y el entorno gráfico es muy amigable, y al igual que las restantes no guarda las notas de los alumnos para ser analizadas por los docentes.

2 En un paquete de domino, hay 28 fichas. Elegimos una ficha al azar del paquete completo. Cuál es la probabilidad de que contenga un 6?

a)  $6/28$                       b)  $1/4$

c)  $1/2$                               d)  $1/7$

**Correctas:0    Incorrectas: 1    No contestadas: 19**  
**NOTA: 0.**

Test finalizado. Puedes revisar tus respuestas ya corregidas.

[< Anterior](#)                      [Siguiente >](#)

**Ilustración 8. Funcionamiento de la aplicación Educarex**

Como conclusión cabe destacar que no hemos hallado ninguna aplicación que cumpla nuestros requisitos más importantes, como que cada alumno se identifique y resuelva las tareas, enviando los resultados al profesor del curso, y sobre todo fomentar el aprendizaje adaptativo, variando el nivel de los ejercicios de acuerdo a como los van resolviendo.

## 2.3 Tecnologías utilizadas para el desarrollo de la aplicación

---

A continuación mostramos las tecnologías que hicieron posible el desarrollo de la aplicación.

Las herramientas utilizadas fueron seleccionadas luego de varias tertulias con los compañeros de curso, quienes nos aconsejaron cuales eran las más sencillas para el desarrollo Web, y que su aprendizaje nos podría generar oportunidades laborales en el futuro, principalmente el lenguaje de programación.

### 2.3.1 Lenguaje de programación: *Ruby*

---



**Ilustración 9. Logo Ruby**

*Ruby* es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en *Ruby* en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en *Python* y *Perl* con características de programación orientada a objetos. Su implementación oficial es distribuida bajo una licencia de *software* libre. El nombre deriva de una broma aludiendo al lenguaje *Perl* (perla) y lo llamaron *Ruby* (en español rubí).

*Ruby* está orientado a objetos: todos los tipos de datos son un objeto, incluidas las clases y tipos. Toda función es un método. Las variables son referencias a objetos. Así mismo soporta herencia con enlace dinámico, aunque por el contrario no soporta herencia múltiple.

*Ruby* soporta polimorfismo de tipos (permite tratar a subclases utilizando la interfaz de la clase padre).

Las principales características de este lenguaje son:

- Orientado a objetos

- Manejo de excepciones
- Altamente portable
- Amplia librería
- En tiempo de ejecución, soporta alteración de objetos

## 2.3.2 Framework Ruby on Rails

---



**Ilustración 10. Logo Ruby on Rails**

*Ruby on Rails*, también conocido como RoR o *Rails*, es un *framework* de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma del patrón Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros *frameworks* y con un mínimo de configuración. *Rails* se distribuye a través de *RubyGems*, que es el formato oficial de paquete y canal de distribución de bibliotecas y aplicaciones *Ruby*.

Las gemas son *plugins* añadidos a nuestros proyectos en *Rails*, que nos permiten nuevas funcionalidades, funciones predefinidas o nuevas herramientas para el desarrollo. El listado de gemas se encuentra en la web de *RubyForge*.

Se basa en el modelo-vista-controlador o MVC (Controlador => Modelo => Vista), que fragmenta la programación en los siguientes aspectos para la implementación de nuestro sistema:

## Modelo

Este miembro del controlador maneja las operaciones lógicas. El Modelo consiste en las clases que representan a las tablas de la base de datos, el acceso a la información y también la actualización de las tablas. Los datos normalmente estarán en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán las correspondientes consultas (*SELECT*, *UPDATE*, *INSERT*).

## Vista

Es la lógica de visualización, o cómo se muestran los datos de las clases del controlador. El método que se emplea en *Rails* por defecto es usar *Ruby* Empotrado (archivos.rhtml, desde la versión 2.x en adelante de RoR archivos.html.erb), que son básicamente fragmentos de código HTML con algo de código en *Ruby*. Le corresponde dibujar, o expresar la última forma de los datos: la interfaz gráfica que interactúa con el usuario final del programa (GUI).

## Controlador

Hace de intermediario entre la vista y el controlador. Contiene el código necesario para responder a las acciones que se realizan en la aplicación, como realizar una compra, hacer un ejercicio, o sacar estadísticas etcétera.

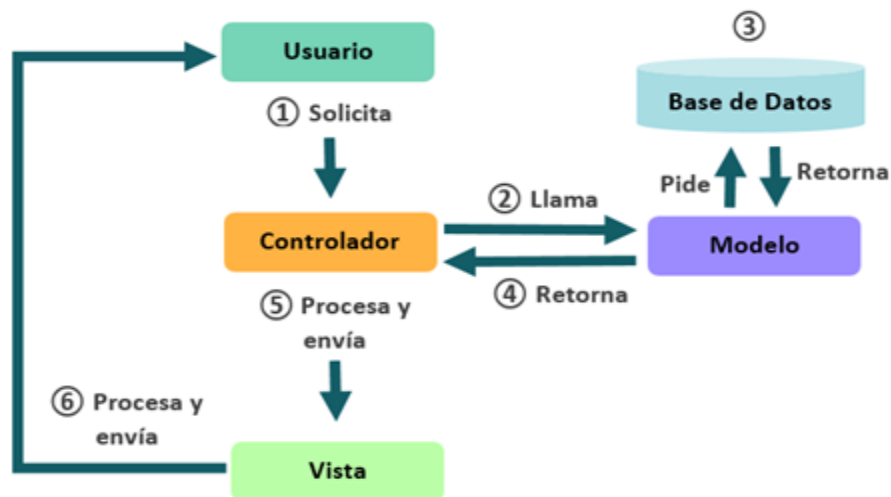


Ilustración 11. Ciclo de vida del MVC

Utilizar un *framework* tiene algunas ventajas, como:

- Facilidad para el programador a la hora de diseñar la estructura global de la aplicación. El *framework* proporciona la estructura, la cual hay que ir completando.

- Facilita la colaboración y el trabajo en equipo, ya que está todo más estandarizado y definido según el *framework*. Es un código menos personal y más automatizado.

- Hay un mayor número de utilidades y librerías adaptadas a *frameworks* en concreto.

- Aunque al desarrollar una aplicación, se necesita un cierto tiempo y costes iniciales de aprendizaje, a largo plazo se facilita el desarrollo de la aplicación y el mantenimiento de la misma.

En este proyecto vamos a utilizar la versión 4.0.0 de *Rails*. Hemos elegido este lenguaje por la facilidad de uso y aprendizaje. Previo aprendizaje de *Ruby*, *Rails* es muy intuitivo y relativamente sencillo, y permite el desarrollo de una web en un intervalo corto de tiempo [7].

### 2.3.3 Base de datos: *SQLite3*

---



**Ilustración 12. Logo SQLite**

SQLite es un sistema de gestión de bases de datos relacional, contenida en una biblioteca escrita en C.

A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos.

En su versión 3, que es la seleccionada para nuestra aplicación, SQLite permite bases de datos de hasta 2 Terabytes de tamaño.

Varios procesos o hilos pueden acceder a la misma base de datos sin problemas. Varios accesos de lectura pueden ser servidos en paralelo. Un acceso de escritura sólo puede ser servido si no se está sirviendo ningún otro



acceso concurrentemente. En caso contrario, el acceso de escritura falla devolviendo un código de error (o puede automáticamente reintentarse hasta que expira un tiempo de expiración configurable).

Características principales:

- *SQLite* soporta múltiples tablas, índices, *triggers* y vistas.
- Lee y escribe directamente sobre archivos que se encuentran en el disco duro.
- Utiliza el espacio en disco que es realmente necesario en cada momento ya que emplea registros de tamaño variable.
- *SQL* realiza operaciones de manera eficiente y es más rápido que *MySQL* y *PostgreSQL*.
- Es totalmente auto contenida es decir que no tiene dependencias externas.
- Cuenta con librerías de acceso para muchos lenguajes de programación.
- Soporta funciones *SQL* definidas por el usuario (UDF).
- El código fuente es de dominio público y se encuentra muy bien documentado

### 2.3.4 Editor de código utilizado: *Sublime Text*

---



**Ilustración 13. Logo Sublime Text**

*Sublime Text* es un editor de código multiplataforma, ligero. Es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente.

*Sublime Text* permite tener varios documentos abiertos mediante pestañas. Dispone de modo de pantalla completa, para aprovechar al máximo el espacio visual disponible de la pantalla.

Es muy útil para desplazarse por el archivo cuando conocemos bien la estructura del mismo, y sobre todo cuándo contiene muchas líneas de código.

Se puede hacer búsqueda de expresiones regulares o normales por archivos, por proyectos, por directorios, por una conjunción de ellos o por todo a la vez.

El aprendizaje de dichas herramientas ha resultado complejo, principalmente el lenguaje de programación dado que debimos empezar por lo más básico, pero la motivación por terminar la carrera y el intento de superación personal han vencido a las adversidades. A su vez contamos con el apoyo de muy buenos desarrolladores compañeros de curso.

En el siguiente capítulo se describen los pasos previos a la implementación de la aplicación que hicieron posible las herramientas mencionadas.

---

# CAPÍTULO 3

---

## Análisis

En este capítulo exponemos el análisis con la descripción de la solución propuesta, mediante la cual se intenta descubrir qué es lo que se necesita y se llega a una comprensión de los requerimientos del sistema, es decir las características que el sistema debe tener.

### 3.1 Análisis de requisitos

---

En esta sección detallaremos los requisitos funcionales y no funcionales asociados al proyecto.

#### Requisitos funcionales

---

El sistema implementado se basa en una aplicación web para utilizar en las escuelas de secundaria y en su primera versión se basa sólo en la enseñanza de las matemáticas.

Existen tres tipos de actores que son los alumnos, los profesores y uno o varios “usuarios administradores”.

El administrador se encarga de dar de alta el curso, los alumnos, y los profesores, modificarlos o eliminarlos. También podrá crear nuevos usuarios de tipo administradores, si lo necesita.

El sistema recoge los datos de una base de datos relacional, manipulable por el administrador y modificable.

Los alumnos inician sesión en el sistema con un usuario que será su correo electrónico y una contraseña. Una vez dentro pueden practicar ejercicios

de cada unidad todas las veces que quieran hasta que incorporen los conocimientos. El sistema dará ejercicios más accesibles cuando detecte que el alumno presenta dificultades, y más difíciles cuando tenga un buen rendimiento. Cuando se sientan capacitados, los alumnos pueden acceder a las tareas, que serán puntuables, y cuya nota será recibida por el docente.

Los alumnos también tienen acceso a estadísticas que muestran todos los resultados de las tareas realizadas y la nota promedio.

Los profesores tienen la misma modalidad de acceso que los alumnos, pero tienen una vista diferente de la aplicación, ya que pueden practicar ejercicios pero no realizan tareas. A su vez los docentes tienen acceso a las estadísticas de todo el curso, por cada alumno, y de cada unidad.

## Requisitos no funcionales

---

La herramienta debe realizar su tarea de una forma fluida garantizando un óptimo rendimiento. Toda funcionalidad debe responder al usuario en menos de cinco segundos.

El sistema debe ser capaz de operar adecuadamente con hasta 1000 usuarios con sesiones concurrentes.

La aplicación web debe poseer un diseño “*Responsive*” a fin de garantizar la adecuada visualización en múltiples computadores personales, dispositivos tableta y teléfonos inteligentes.

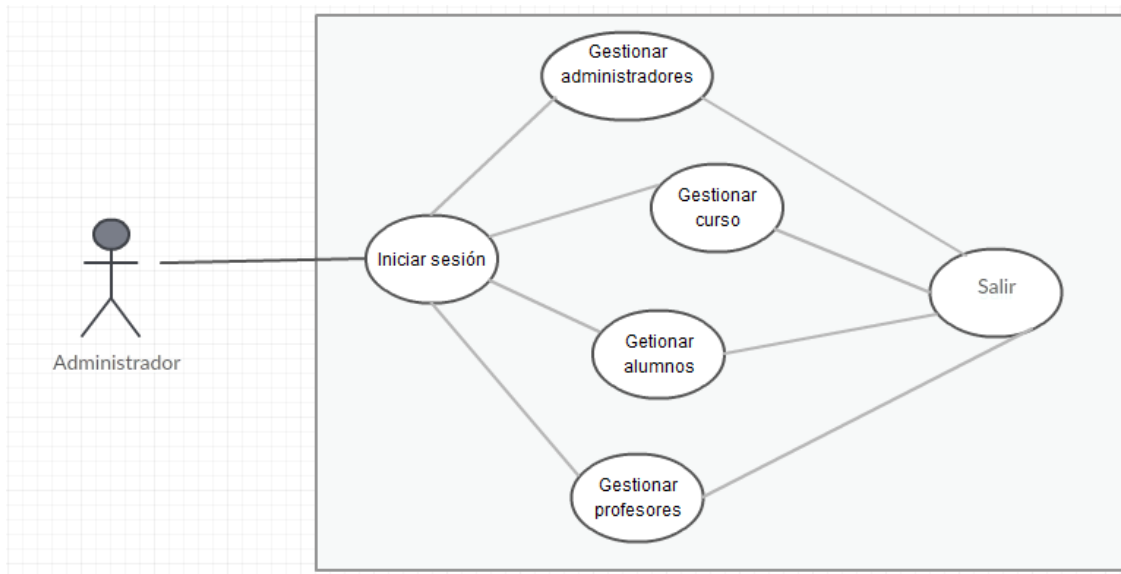
El tiempo de aprendizaje del sistema por un usuario deberá ser menor a una hora.

## 3.2 Casos de uso

---

De los requisitos explicados en el punto anterior se extraen los siguientes casos de uso:

### Caso de uso del administrador al iniciar sesión



**Ilustración 14. Casos de uso del administrador**

## Caso de uso del alumno al iniciar sesión

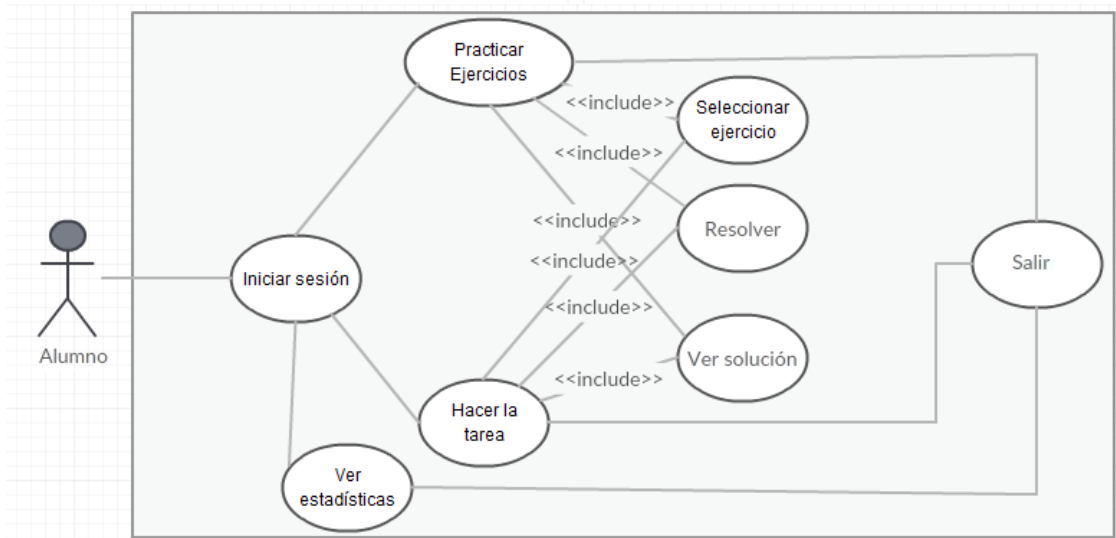


Ilustración 15. Casos de uso del Alumno

## Casos de uso del profesor al iniciar sesión

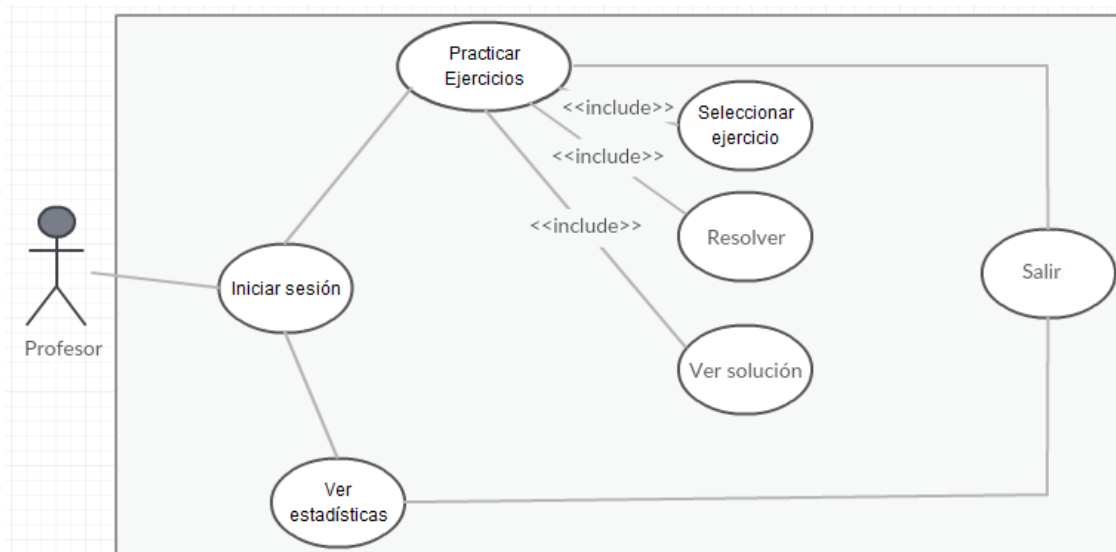


Ilustración 16. Casos de uso del profesor

### 3.3 Diagrama de clases

El diagrama de clases que describe las entidades del proyecto es el que sigue:

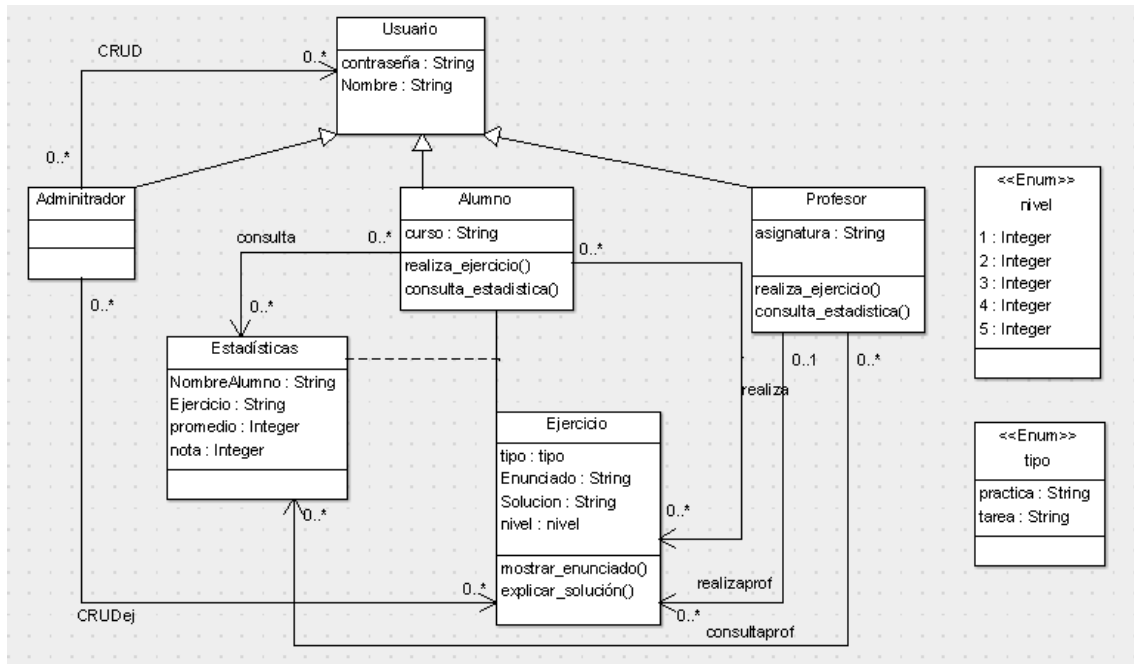


Ilustración 17. Diagrama de clases

La fase de análisis es esencial, para evitar el fracaso del proyecto, dado que si no sabemos con precisión qué es lo que se necesita, ningún proceso de desarrollo nos permitirá obtenerlo.

En el siguiente capítulo se detallarán los aspectos de diseño seleccionados para la implementación de la aplicación.





---

# CAPÍTULO 4

---

## Diseño

Mientras que en la etapa de análisis se definieron los requisitos del usuario desde distintos puntos de vista (el qué), en la fase de diseño se representan las características del sistema que nos permitirán implementarlo de forma efectiva (el cómo). Este trabajo consiste en dos fases: realizar un estudio del contexto matemático en el que encuadrar la aplicación, y diseñar una estructura de archivos que permita el desarrollo.

### 4.1 Contexto matemático

---

Tal y como se ha comentado en el resumen inicial, la aplicación está dirigida para ser utilizada por alumnos y profesores de la ESO. Se hace necesario por lo tanto, un estudio de los apartados concretos en los que enmarcar el trabajo.

Tras el estudio del temario impartido en los niveles en los que se pretende encuadrar la aplicación, se decide crear un sistema con ejercicios pertenecientes a primer año de la ESO (véase apéndice 2).

Las unidades seleccionadas para esta primera versión de la aplicación son las siguientes:

- Enteros, números negativos y potencias.
- Números decimales, fracciones y raíces.
- Cálculos con fracciones.

## 4.2 Estructura de la aplicación

---

Una vez delimitado el ámbito concreto de la aplicación y conocidas las tecnologías con las que se implanta la misma, es el momento de organizar la estructura a utilizar.

Como se ha descrito en el capítulo 2, utilizaremos el *framework Ruby on Rails*, el cual está compuesto de los siguientes archivos y directorios bajo la arquitectura MVC:

app: Contiene, los controladores, vistas, modelos, *helpers* y *assets*, la mayor parte del trabajo lo realizaremos aquí.

config: Contiene archivos de configuración de nuestra aplicación, tales como bases de datos y rutas.

db: Contiene información de nuestro esquema actual de la base de datos o bien el archivo con extensión *sqlite*, las migraciones y los *seeds* (scripts que nos permiten poblar nuestra base de datos, con datos para prueba).

lib: Contiene librerías que nos permiten extender nuestra aplicación.

log: Contiene registros de la aplicación como las peticiones del usuario, etc.

public: Contiene archivos estáticos y nuestros *assets* compilados.

test/spec: Contiene archivos con pruebas unitarias y funcionales de nuestra aplicación.

tmp: Contiene archivos temporales, tales como caché, archivos de sesión, etc.

vendor: Contiene los archivos de librerías externas.

Gemfile/Gemfile.lock: Estos archivos contienen todas las dependencias necesarias para correr nuestra aplicación.

## 4.3 Características intrínsecas del diseño

---

En el presente apartado, se detallan, las características más importantes sobre las que se trabajó en el diseño para cumplir con los requerimientos del sistema.

Ruby cuenta con un generador de números aleatorios: el método `rand`. Usando `rand` se obtiene un número aleatorio  $x$ , tal que  $0 \leq x < 1$ . Si se le da un parámetro, por ejemplo `rand(5)`, entonces se obtiene un número entero

entre 0 y 4 (ambos incluidos). Por medio de este método, definiremos 5 niveles distintos de ejercicios. Los alumnos comenzarán ejercitando en el nivel 3, y luego de dos fallos, pasarán al nivel 2, o por el contrario, luego de dos aciertos, pasarán al nivel 4.

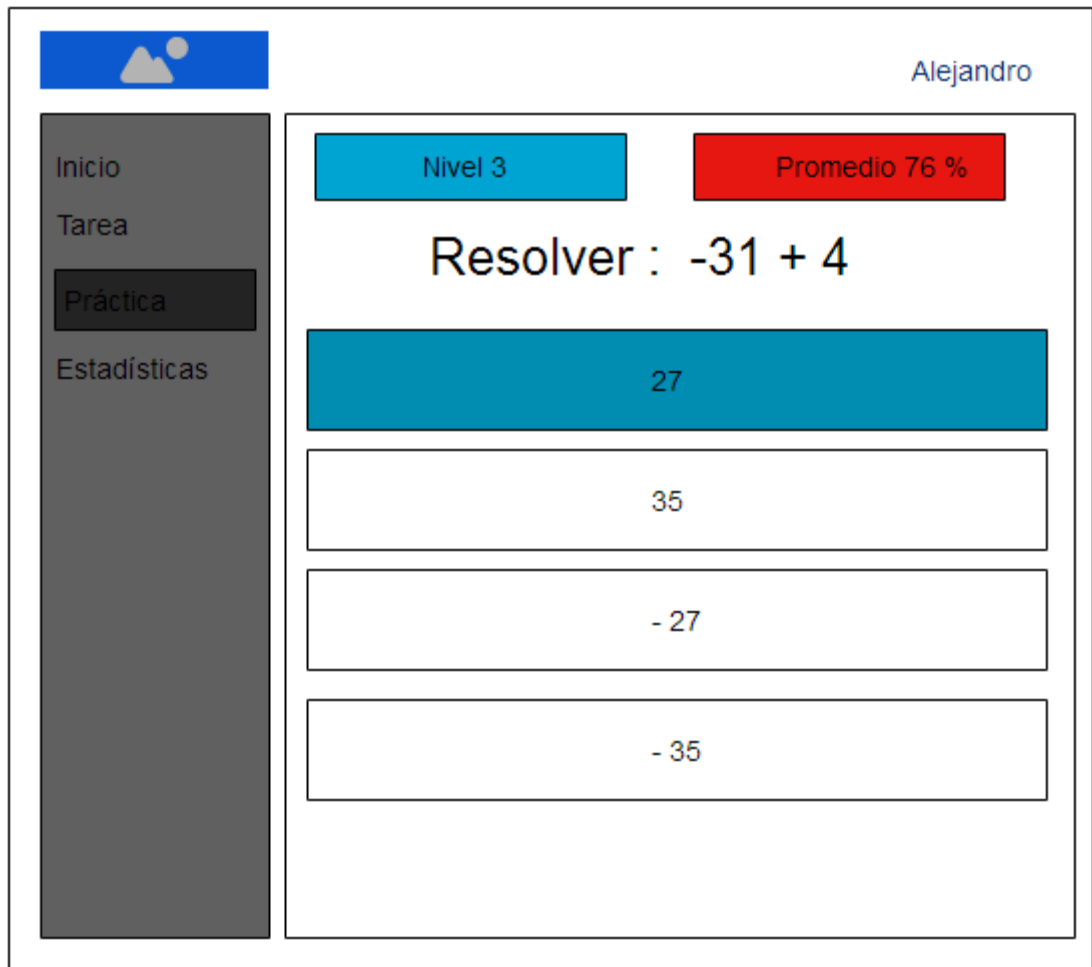
Los ejercicios se plantean con la metodología *Múltiple Choice*, por lo que los alumnos recibirán junto con el enunciado del ejercicio, cuatro opciones posibles de respuesta, entre las que deberán seleccionar una.

Las tareas que realizan los alumnos, cuyas notas serán enviadas al profesor, tendrán un nivel tres predefinido.

## 4.4 Prototipo de la interfaz gráfica de usuario

El siguiente prototipo, se basa en el usuario alumno, resolviendo un ejercicio de sumas y restas de enteros. Decidimos mostrar sobre la izquierda el menú, y a la derecha el ejercicio seleccionado, para no perder de vista donde nos encontramos.

A su vez en el vértice superior derecho mantenemos activo durante toda la sesión el nombre del usuario.



**Ilustración 18. Prototipo de la interfaz gráfica de usuario**

La etapa de diseño ha determinado ¿cómo debe ser construido el sistema?, y una vez seleccionado el lenguaje más adecuado y el sistema gestor de Bases de Datos a utilizar se procede, como detallamos en el próximo capítulo a la implementación de la aplicación.

---

# CAPÍTULO 5

---

## Implementación

En este capítulo se detalla cómo se implementaron las funciones más destacadas de la aplicación y se refleja el código utilizado para resolverlas y posteriormente presentamos breve descripción de las herramientas utilizadas para el diseño gráfico.

### 5.1 Aprendizaje adaptativo

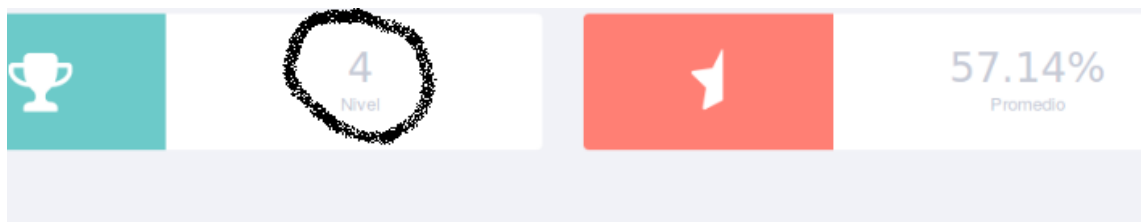
---

Cuando el alumno realiza prácticas, los ejercicios se adaptan a su nivel, haciéndose más difíciles cuando vaya respondiendo correctamente, o más fáciles cuando se equivoca de manera reiterada.

A continuación se puede apreciar el ejemplo de las sumas y restas de números enteros utilizando el método *rand*, el cual se explicó en la fase de diseño, al cual se le aplican distintos parámetros de acuerdo al nivel en el que se encuentra el alumno practicando ejercicios:

```
if level == 1
    param1 = rand(-5..-1)
    param2 = rand(3..10)
elseif level == 2
    param1 = rand(-5..-1)
    param2 = rand(12..39)
elseif level == 3
    param1 = rand(-39..-8)
    param2 = rand(12..39)
elseif level == 4
```

```
param1 = rand(-39..-8)
param2 = rand(-5..-1)
elsif level == 5
  param1 = rand(-39..-8)
  param2 = rand(-39..-8)
end
```



**Resolver: -31 + -4**

**Ilustración 19. Cambio de nivel de ejercicio**

## 5.2 Posibles respuestas

El alumno puede elegir entre cuatro respuestas posibles, de las cuales solo una puede ser la correcta. Para ello se genera una respuesta correcta y tres erróneas agregando algún signo, o sumando en lugar de restar como puede apreciarse en el siguiente código.

```
def generate_correct_answer(args)
  "#{args[0] + args[1]}".html_safe
end
```

```
def generate_wrong_answer_1(args)
  ans = "#{(args[0]*-1) + args[1]}"
```

```
ans.html_safe  
end
```

```
def generate_wrong_answer_2(args)  
  ans = "#{args[0] - args[1]}"  
  ans.html_safe  
end
```

```
def generate_wrong_answer_3(args)  
  ans = "#{(args[0]*-1) - args[1]}"  
  ans.html_safe  
end
```

---

**Resolver: -31 + -4**

**27**

35

-27

**Ilustración 20. Posibles respuestas**

## 5.3 Explicación de respuesta incorrecta

---

Cuando el alumno se equivoca resolviendo un ejercicio el sistema le explica cómo debería haberlo resuelto al momento, de esta manera garantizamos un aprendizaje más efectivo que si tuviera que esperar a que el docente se lo corrija en una hoja.

En el siguiente fragmento de código podemos apreciar cómo se resolvió la cuestión, llamando a la variable “*explanation*”, la cual contiene la explicación para este ejercicio, y con las variables que el método *rand* dio en este caso concreto.

```
def generate_explanation(args)
  "<h3>#{[18n.t('exercises.numbers.change_to_decimals.explanation',
args0:      args[0],      args1:      args[1],      answer:
generate_correct_answer(args))}</h3>".html_safe

  End

//variables de la carpeta config/locales

change_to_decimals:

  solve: "Convertir la siguiente fracción en un número decimal:
  %{fraction}"

  explanation: "Para resolver este problema necesitamos dividir el
  numerador de la fracción por el denominador. <br><br> En este caso:
  %{args0} &divide; %{args1} = %{answer}"
```





Ilustración 21. Explicación de respuesta incorrecta

## 5.4 Panel profesor

En este caso podemos ver el panel del profesor para el cual imprimimos por pantalla una tabla con todos los alumnos y el promedio general de cada uno para saber quiénes son los alumnos aprobados.

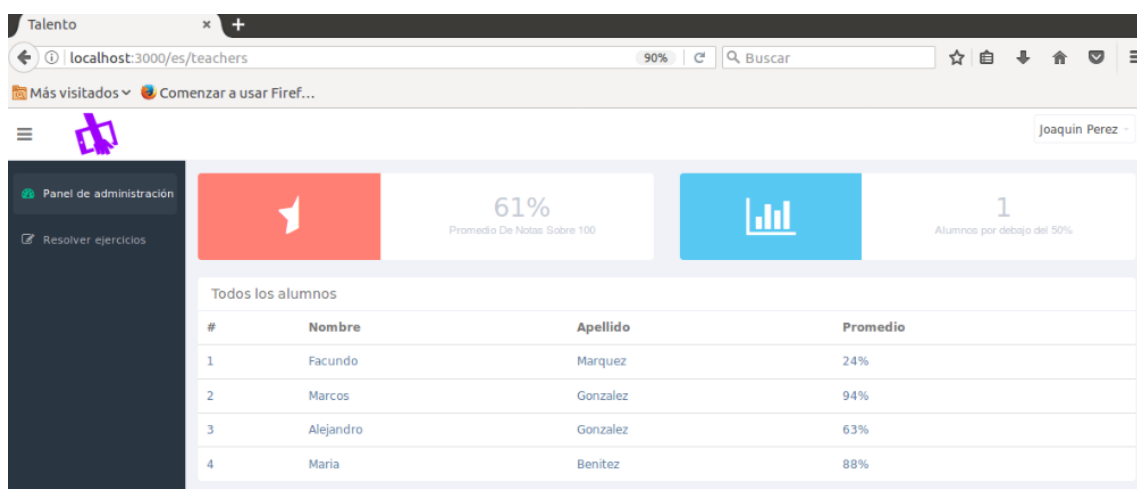


Ilustración 22. Panel profesor

## 5.5 Diseño gráfico de la aplicación

---

El diseño gráfico ha sido llevado a cabo utilizando las tecnologías CSS y *Bootstrap*. Esta última, es una excelente herramienta para crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño. Además, *Bootstrap* ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus librerías.

Se ha logrado una interfaz amigable, que mantiene la misma línea en todas sus páginas, y que ha sido probada solo en un ordenador de mesa, por lo que por el momento no ha sido posible probar si se adapta a otros dispositivos.

Para lograrlo se han realizado las siguientes tareas:

- Agregar en la cabecera de las páginas las siguientes etiquetas:

```
<link rel="stylesheet" href="http://netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/css/
bootstrap-combined.min.css" />
<link rel="stylesheet" href="../../static/stylesheets/bootstrap-switch.css" />
<link rel="stylesheet" href="http://getbootstrap.com/2.3.2/assets/js/google-code-prettify/
prettify.css" />
<link rel="stylesheet" href="http://getbootstrap.com/2.3.2/assets/css/docs.css" />
<link rel="stylesheet" href="http://netdna.bootstrapcdn.com/font-awesome/3.2.1/css/
font-awesome.css">
<link rel="stylesheet" href="http://bdmdesign.github.io/bootstrap-switch/static/stylesheets/
flat-ui-fonts.css">
```

- Introducir en cada página una cabecera con las siguientes características:
  - Título con el nombre de la aplicación.
  - Logo de la aplicación
  - En los casos en los que es necesario, botones de navegación.
- Con respecto a los estilos se ha introducido en cada elemento HTML la clase Bootstrap que se quisiese representar, a través del atributo "class".
- Para la distribución de los elementos en la pantalla se ha utilizado la etiqueta "<div>" [8].

En esta fase se han codificado los algoritmos alcanzando los objetivos planteados, a continuación se desarrollarán las conclusiones a las que arribamos.

---

---

# CAPÍTULO 6

---

## Conclusiones

En este capítulo incluimos una sección en la que describimos los diferentes problemas y contratiempos que hemos sufrido durante el transcurso del proyecto. Además, se plantean las consideraciones finales obtenidas, realizando un resumen final, y analizando si se ha logrado cumplir con los objetivos propuestos al comienzo del proyecto.

### 6.1 Dificultades encontradas

---

A lo largo de toda la implementación de esta primera versión de la aplicación han surgido innumerables contratiempos puesto que para su creación ha sido necesario estudiar programas que no se han utilizado en la carrera. Para ello hemos contado con el apoyo incondicional de Cristian Cañadas, excelente desarrollador y compañero en este último curso, resolviendo dudas, y ayudándonos a solucionar conflictos que surgieron a lo largo de la implementación.

En un principio debimos familiarizarnos con el lenguaje *Ruby*, y allí surgió el primer inconveniente ya que debimos instalar el sistema operativo *Ubuntu*, para lograr una adecuada conexión con el servidor.

El diseño de la base de datos fue sufriendo cambios a lo largo de todo el proceso dado que no podíamos cumplir con las funcionalidades requeridas, como creación de nuevas clases o agregación de atributos.

La implementación ha significado un gran esfuerzo, ya que tardó más de tres meses en terminarse, dado que cada nueva función requería un amplio estudio y cientos de casos de prueba y error.

## 6.2 Consideraciones finales

---

Mediante este proyecto pretendemos alentar a la comunidad educativa a descubrir si la educación virtual puede mejorar la forma en la que se enseña y el modo en que los estudiantes aprenden, impulsando la tecnología como instrumento mediador que facilita el aprendizaje en la distancia posibilitando nuevos tipos de transmisión del conocimiento.

La aplicación implementada en el presente proyecto cumple con los objetivos propuestos, liberando al docente de la carga que supone la corrección de tareas y optimizando el aprendizaje del alumnado, recibiendo respuestas y explicaciones inmediatas, y adaptando el nivel de dificultad de los ejercicios al nivel de cada alumno.

A su vez, el docente, sólo con iniciar sesión, tendrá toda la información de los alumnos, quiénes realizaron bien o mal la tarea y qué temas deberían reforzarse.

Desde la educación, necesitamos emprender iniciativas innovadoras, para aprovechar las enormes posibilidades que para las tareas de enseñanza proporcionan las TIC. Y como tal, toda innovación es arriesgada y requiere un compromiso a largo plazo para superar las resistencias inevitables por parte de los agentes conservadores [5].

## 6.3 Líneas futuras

---

Con respecto a los puntos en los que seguiremos trabajando, surgen varias ideas, sobre todo en nuevas funcionalidades:

- Completar con ejercicios todas las tareas del curso de primero de la ESO y a continuación de los siguientes años.
- Una función para el docente que genere los exámenes en formato PDF.
- Variedad de consultas para los docentes, sobre unidades, ejercicios y alumnos.  
Por ejemplo:
  - ¿quiénes no aprobaron el ejercicio de sumas y restas?
  - ¿cuántos ejercicios practicaron los alumnos que no aprobaron?
- Proporcionar un teclado en pantalla para introducir las soluciones.
- Ampliar la aplicación a otras áreas de estudio como la Física o la Química

---

## Bibliografía

- [1] S. V. C. Breitzler, «BuscaPic , detección de picos mediante realidad aumentada y el servicio Goggle Maps Elevation API», Trabajo Final Grado, Universidad Politécnica de Valencia, 2016.
- [2] Á. R. Coves, «Diseño e implementación de una máquina Arcade con Raspberry Pi y Arduino», Trab. Final Grado, Universidad Politécnica de Valencia, 2015.
- [3] I. Bautista Perales, «Aplicación Web de bases de datos usando el Framework Ruby on Rails», Trab. Final Grado, Universidad Politécnica de Valencia, 2014.
- [4] A. Almirón García, «Aplicación web para enseñanza/aprendizaje de cálculo y álgebra», Trab. Final Grado, Universidad de Málaga, 2015.
- [5] M. Muñoz-Vázquez, «El Docente de Educacion Virtual. Guia básica», *Comun. VO - 40*, 2013.
- [6] A. Garrido, «<http://edmodo.antoniogarrido.es/index.html>». [Último acceso: 06-07-2017]
- [7] M. Hartl, *Ruby on Rails 3 Tutorial*. 2011. <https://www.railstutorial.org/book> [Último acceso: 28-06-2017]
- [8] B. Efron y R. LePage, «Introduction to bootstrap», *Explor. limits bootstrap*, 1992.



---

# APÉNDICE 1

---

## Imágenes de la primera versión de Talento



Ilustración 23. Pantalla de inicio



Ilustración 24. Nuestra aplicación



Ilustración 25. Contacto

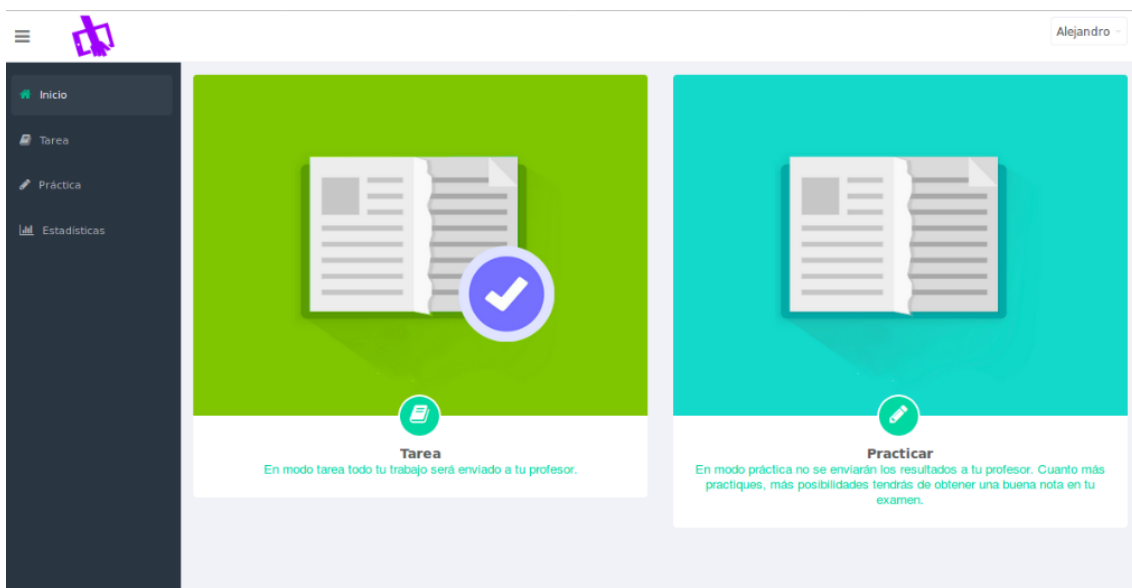


Ilustración 26. Inicio estudiantes



The screenshot shows a user interface for a student named Alejandro. On the left is a dark sidebar with navigation options: Inicio, Tarea, Práctica, and Estadísticas. The main area is titled 'Temas Disponibles' and contains three topic cards:

- N1. Unidad 1:**  $4 + (-5) = -1$  and  $(-7) + 11 = 4$
- N2. Unidad 2:**  $0,7 = \frac{7}{10}$
- N3. Unidad 3:**  $\frac{5}{7} \times \frac{6}{13}$

Each card has a green pencil icon at the bottom, indicating that tasks are available for these topics.

Ilustración 27. Alumno, tareas

The screenshot shows the 'Estadísticas' (Statistics) section for the same student, Alejandro. The main area is titled 'Todos los temas' and displays a table with the following data:

Tema	Promedio	Estado
Unidad 1	62%	Completado
Unidad 2	68%	Completado
Unidad 3	53%	Completado

The 'Estado' column uses green buttons with a checkmark icon to indicate that all three units have been completed.

Ilustración 28. Alumno, estadísticas

# Aplicación web para la enseñanza y aprendizaje de matemáticas

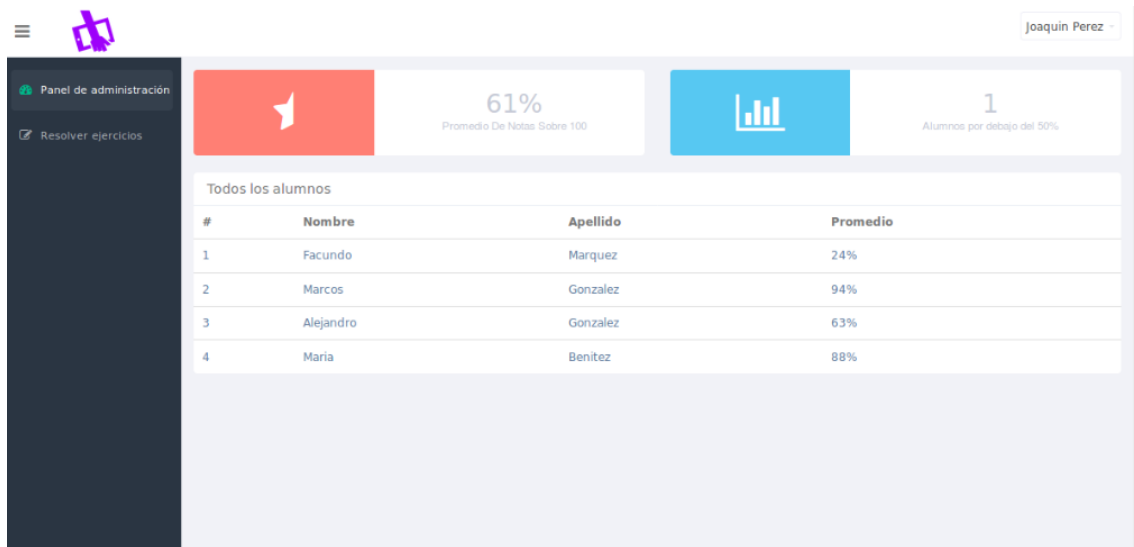


Ilustración 29. Profesores, panel administración

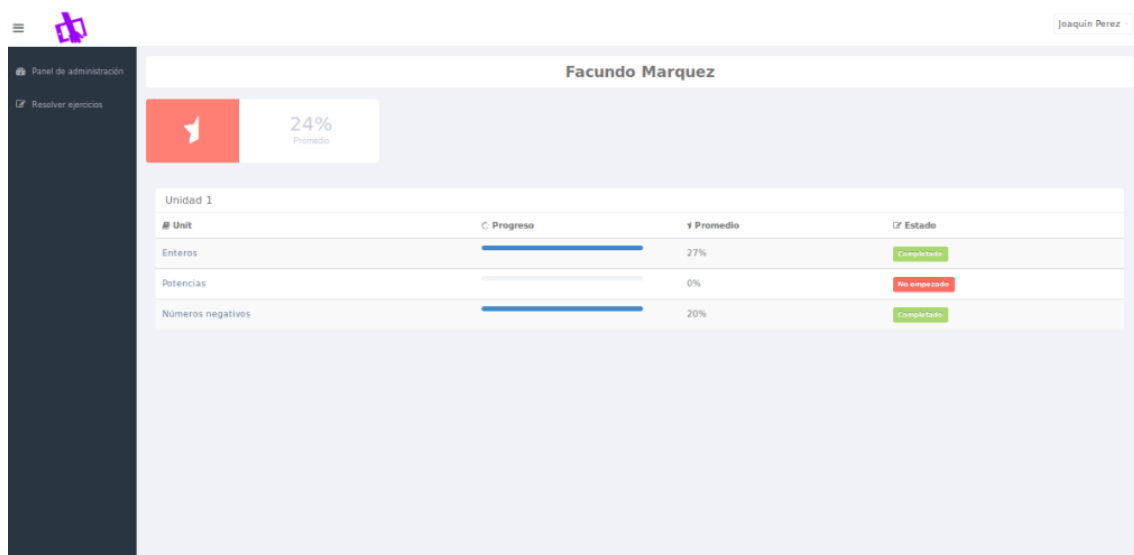


Ilustración 30. Estadísticas de un alumno en concreto

---

## APÉNDICE 2

---

### Objetivos didácticos específicos de 1º de E.S.O (Establecidos por la consejería de educación)

Unidad 1: Los números naturales.

Expresa oralmente y por escrito los conceptos, procedimientos y terminología de los números naturales con propiedad.

Identifica los números naturales y los descompone.

Representa en la recta números naturales.

Ordena números naturales.

Realiza correctamente sumas, restas, multiplicaciones y divisiones con números naturales y utiliza sus propiedades.

Aplica correctamente la jerarquía de las operaciones con operaciones combinadas.

Resuelve problemas aritméticos con números naturales.

Unidad 2: Los números enteros.

Utiliza los números negativos para expresar información de la vida cotidiana.

Identifica el conjunto de los números enteros como una clase que incluye al conjunto de los números naturales.

Representa gráficamente números enteros.

Calcula el valor absoluto de un número entero.

Ordena números enteros.

Realiza correctamente sumas, restas, multiplicaciones y divisiones con números enteros.

Aplica correctamente la jerarquía de las operaciones con operaciones combinadas.

Resuelve problemas para los que se precise la utilización de los números enteros.

Elige la forma de cálculo apropiada (mental, por escrito, con calculadora o con ordenador) y valora la adecuación del resultado al contexto.

### Unidad 3: Divisibilidad

Expresa oralmente y por escrito los conceptos, procedimientos y terminología de la divisibilidad con propiedad.

Identifica y utiliza la relación “ser divisor de” y “ser múltiplo de”.

Reconoce con soltura los primeros números primos (hasta 99).

Identifica con soltura cuándo un número es divisible entre 2, 3 y 5

Descompón un número en factores primos con corrección.

Calcula el máximo común divisor de dos o más números.

Calcula el mínimo común múltiplo de dos o más números.

Elige la forma de cálculo apropiada: mentalmente, por descomposición en factores primos o con ordenador, del máximo común divisor o el mínimo común múltiplo; y valora la adecuación del resultado al contexto en la resolución de situaciones problemáticas.

### Unidad 4: Las fracciones.

Utiliza las fracciones para intercambiar información de la vida cotidiana.

Identifica y usa las fracciones como división, como parte de una unidad y como un operador.

Representa fracciones en la recta numérica.

Identifica las fracciones impropias.

Reconoce fracciones equivalentes.

Reduce fracciones a común denominador.

Ordena fracciones de menor a mayor y viceversa.

Obtiene la fracción irreducible de una fracción dada.

Suma y resta fracciones con el mismo denominador y con distinto denominador.

Multiplica y divide fracciones.

Opera con corrección y utilizando la jerarquía de las operaciones en operaciones combinadas con fracciones.

Resuelve problemas aritméticos con fracciones y elige la forma de cálculo apropiada y valora la adecuación del resultado al contexto.

#### Unidad 5: Los números decimales.

Utiliza los números decimales para intercambiar información de la vida cotidiana.

Transforma unidades.

Descompone un número decimal.

Identifica una fracción decimal y expresa un número decimal exacto como una fracción.

Representa números decimales en la recta.

Ordena números decimales.

Realiza correctamente operaciones con decimales.

Aplica correctamente la jerarquía con operaciones combinadas con decimales.

Redondea a una determinada cifra decimal.

Resuelve problemas aritméticos con decimales.

Elige la forma de cálculo apropiada (mentalmente, por escrito, con calculadora o con ordenador) y valora la adecuación del resultado al contexto.

#### Unidad 6: Introducción al álgebra.

Escribe en lenguaje algebraico situaciones enunciadas en lenguaje natural.

Resuelve mentalmente ecuaciones sencillas e identifica las equivalentes.

Resuelve ecuaciones con coeficientes enteros sin denominadores.

Resuelve ecuaciones con coeficientes enteros con denominadores.

Resuelve problemas de ecuaciones.

#### Unidad 7: Proporcionalidad numérica.

Identifica una razón y una proporción y calcula un cuarto y un medio proporcional.

Resuelve problemas con magnitudes directa e inversamente proporcionales.

Calcula el tanto por ciento de una cantidad y cantidades sobre las que se ha calculado el tanto por ciento.

Resuelve problemas de descuento y de aumentos porcentuales.

#### Unidad 8: Funciones y gráficas.

Sitúa un punto en el sistema de coordenadas cartesianas en el plano.

Completa una tabla y encuentra la función correspondiente.

Interpreta una gráfica.

#### Unidad 9: Elementos en el plano.

Conoce los principales elementos geométricos.

Resuelve cálculos aritméticos y gráficos con ángulos

Conoce las relaciones entre ángulos y rectas.

#### Unidad 10: Figuras planas. Simetría

Clasifica los triángulos.

Construye triángulos

Clasifica los polígonos.

Reconoce la simetría.

#### Unidad 11: Perímetro y área

Conoce el concepto de perímetro y de área.

Sabe efectuar cambios de unidades.

Calcula áreas y perímetros.

#### Unidad 12: Estadística

Distingue los diferentes tipos de variables.

Construye e interpreta tablas.

Construye e interpreta gráficas estadísticas.

Calcula e interpreta los parámetros estadísticos

.

#### Unidad 13: Probabilidad.

Determina los acontecimientos elementales.

Calcula probabilidades.

Interpreta diagramas de árbol.

.