

Task Vault

Proxecto DAW 2024

Estudiante: Ricardo Otero González

O Grove, 14 de Marzo de 2024.

ÍNDICE

1. Introducción.....	3
2. Descrición Xeral.....	4
3. Planificación do proxecto.....	5
3.1. Análise.....	6
3.2. Deseño.....	8
3.2.1. Xestor de bases de datos: MySQL.....	9
3.2.2. Editor de código: Visual Studio Code.....	12
3.2.3. Linguaxes de programación: PHP e JavaScript.....	12
3.2.4. Framework de estilos da interface de usuario: Bootstrap.....	12
3.2.5. Outras tecnoloxías empregadas.....	13
Composer.....	14
tinyMCE.....	15
4. Instalación de proxecto.....	16
5. Custos do proxecto.....	19
Custos de infraestrutura.....	19
Custos de desenvolvemento e mantemento.....	20
6. Melloras.....	20
7. Manual de usuario.....	21
7.1. Panel de administrador.....	22
7.1.1. Pestana datos de usuario.....	22
7.1.2. Pestana xestión de usuarios.....	23
Crear usuarios.....	24
7.1.3. Pestana xestión de cursos e materias.....	25
7.2. Panel de profesor.....	26
7.2.1. Pestana datos de usuario.....	27
7.2.2. Pestana xestión de alumnos.....	27
Crear alumnos.....	28
7.2.3. Pestana xestión de cursos e materias.....	29
7.2.4. Pestana de xestión de exercicios.....	30
Crear exercicios.....	30
Editor de enunciados.....	31
7.3. Panel de estudante.....	32
7.3.1. Pestana datos de usuario.....	32
7.3.2. Pestana de exercicios.....	33
8. Bibliografía.....	35

1. Introducción

A idea de facer este proxecto xorde do problema que hai en calquera sistema de traballo no que habitualmente se traballe con información que sexa reutilizable, hai unha necesidade de ter dita información xestionada dalgunha maneira, o que supoñerá entre moitas cousas un aforro de tempo e diñeiro.

Neste aspecto, as academias de estudos ou academias de clases particulares, son un de tantos exemplos nos que unha boa xestión da información pode supor un aforro importante sobre todo en tempo.

Task Vault é unha aplicación web que ten como obxectivo poder xestionar dunha maneira simple e organizada os exercicios cos cales traballan as academias de clases particulares. Por un lado, permitir os profesores poder crear, editar e gardar os exercicios para logo poder listalos de acordo cuns criterios de busca. E por outro lado, o sistema tamén esta pensado para que os alumnos poidan ver, imprimir ou descargar os exercicios que os profesores consideren oportunos.

Esta aplicación está orientada cara as pequenas academias de estudos ou de cara os profesores autónomos que axudan coas pasantías os seus alumnos a reforzar os coñecementos impartidos nos estudos oficiais.

Tendo en conta as características anteriores cabe destacar que se busca desenvolver unha aplicación que sexa fácil de usar e fácil de aprender, rápida e segura.

2. Descrición Xeral

Como se comentou anteriormente *Task Vault* é unha aplicación web que ten como obxectivo principal axudar as pequenas academias de clases particulares, proporcionándolles un sistema de almacenaxe e organización para os exercicios que estas academias usan habitualmente.

A aplicación ten tres tipos de usuarios diferentes ou roles: administrador, profesor e estudante. Cada un deles con distintos privilexios de acceso e edición dos datos que garda.

Antes do acceso todos os usuarios teñen que iniciar sesión con un nome de usuario único e un contrasinal, estas credenciais son asignadas polos administradores no momento que dan de alta a conta de un usuario, podendo ser cambiadas con posterioridade tanto polos administradores como polo propio usuario unha vez entra na aplicación.

Unha vez iniciada a sesión todos os usuarios poden modificar certos datos do seu perfil, segundo o seu rol terán mais ou menos restricións. Como era de esperar, os administradores teñen a facultade de eliminar ou xestionar a activación/desactivación das contas de usuario. Tamén teñen a autoridade de modificar calquera información asociada á conta dun usuario, incluído o seu rol.

Os profesores tamén teñen a posibilidade de rexistrar e modificar as contas de usuarios dos estudantes. Os administradores poden engadir, eliminar e editar cursos académicos e materias asociadas a cada un destes cursos. Logo estes cursos e materias serán asociadas a cada estudante segundo se dan de alta as súas contas.

A aplicación conta cun editor de exercicios onde os profesores poden crear, editar e eliminar enunciados de exercicios. Todos os exercicios están sempre dispoñibles para os profesores pero estes poden restrinxir a súa visibilidade para os estudantes segundo conveña. Estes exercicios están asociados a unha materia de un curso determinado, podendo acceder a eles

todos os a estudantes que estuden esa materia, sempre e cando o exercicio sexa activado como visible por un profesor.

O editor de exercicios é un editor de texto enriquecido, onde algunha das súas características son a posibilidade de descarga e impresión dos enunciados, edición de ecuacións matemáticas ou poder ver o editor a pantalla completa para unha maior comodidade.

Tanto usuarios como exercicios poden ser listados e filtrados por certos criterios, o que nos proporciona unha rápida busca dalgún recurso que precisemos nun momento dado.

3. Planificación do proxecto

O modelo de ciclo de vida elixido para o desenvolvemento do proxecto foi o modelo en “cascada con realimentación”. Este modelo é un dos mais utilizados e provén do “modelo en cascada”, pero engadindo a realimentación entre etapas de tal xeito que poidamos volver atrás para corrixir, modificar ou depurar calquera erro detectado.

Elixíuse esta metodoloxía de desenvolvemento xa que é un proxecto pequeno no cal o tempo de desenvolvemento non será moi longo e os requisitos do sistema son bastante claros, o proxecto en principio será pouco evolutivo e con poucos cambios.

3.1. Análise

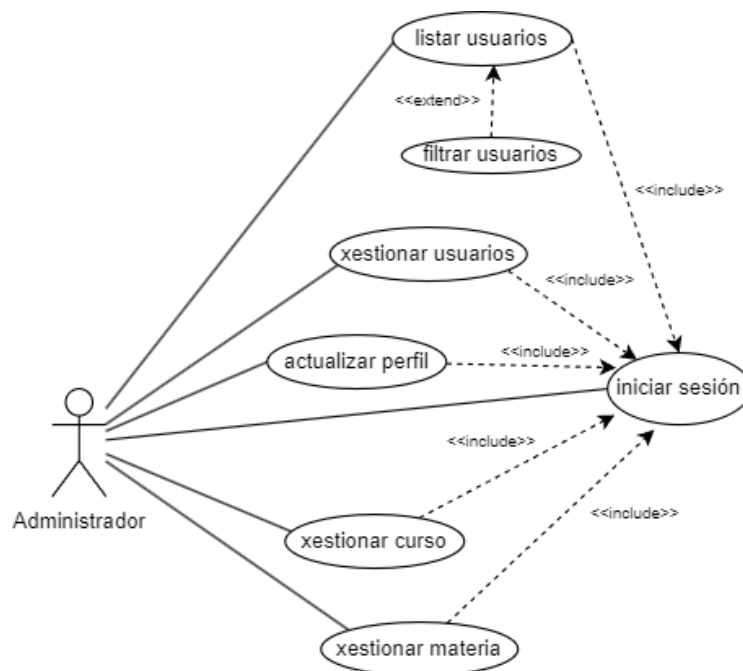
O primeiro que debemos facer é crear unha lista de requisitos en base as necesidades do noso cliente, para iso confiamos na información obtida das reunións previas co cliente días antes de iniciar o desenvolvemento do proxecto. Neste caso ao non contar cun cliente final os requisitos teremos que detallalos nos segundo as necesidades do sistema que queremos desenvolver. A continuación, pasamos a presentar aqueles requisitos que consideramos fundamentais.

Requisitos funcionais:

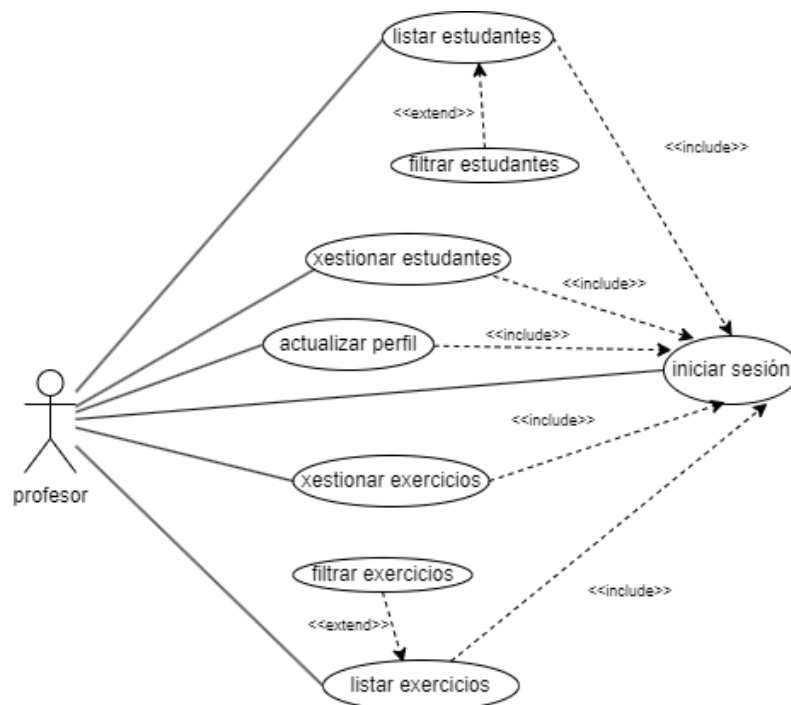
- O sistema terá tres tipos de usuarios: administradores, profesores e estudantes. Cada un deles terá uns privilexios respecto a que información terán dispoñible, e que parte desa información poden modificar.
- Os usuarios deben poder iniciar sesión e pechar sesión.
- O sistema debe permitir os administradores crear, modificar ou eliminar usuarios.
- Os profesores poderán crear, editar e eliminar alumnos e asociarlle as materias que estean estudando.
- O sistema debe permitir os administradores crear, modificar ou eliminar cursos e materias. Cada materia irá asociada a un curso.
- O sistema debe permitir aos profesores crear, ver, actualizar e eliminar exercicios. Os exercicios estarán asociados a un curso e materia determinada.
- Un exercicio pode ser visible ou non para os alumnos segundo o decida o profesor.
- O sistema debe permitir listar os exercicios, e filtralos en función de determinados criterios.
- Os alumnos poderán acceder a lista de exercicios que o profesor lles habilite como visibles.
- O sistema debe permitir a descarga e impresión dos enunciados dos exercicios.
- Todas as distintas listas que mostra a aplicación deberán estar paxinadas para non cargar todos os datos de unha soa vez.
- Todos os usuarios poderán actualizar datos do seu perfil de usuario (segundo o seu rol terán mais ou menos acceso a certa información) .

Requisitos non funcionais:

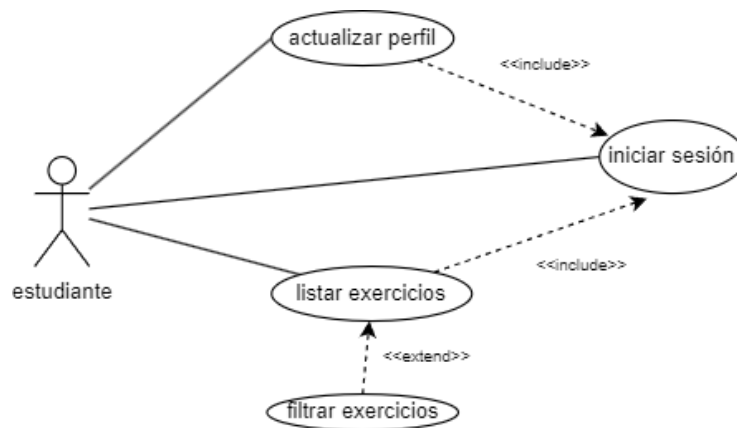
- O sistema debe ser fácil de usar e comprender.
- O sistema debe ser rápido.
- O sistema debe ser seguro.

Diagramas de casos de uso:

Imaxe 2.1: casos de uso dun administrador



Imaxe 2.2: casos de uso dun profesor



Imaxe 2.3: casos de uso dun estudante

3.2. Deseño

O deseño da aplicación web fíxose baixo o patrón de arquitectura *Modelo-vista-controlador* (*MVC*), este é un patrón de arquitectura de software que propón a creación de tres compoñentes ou capas independentes entre si:

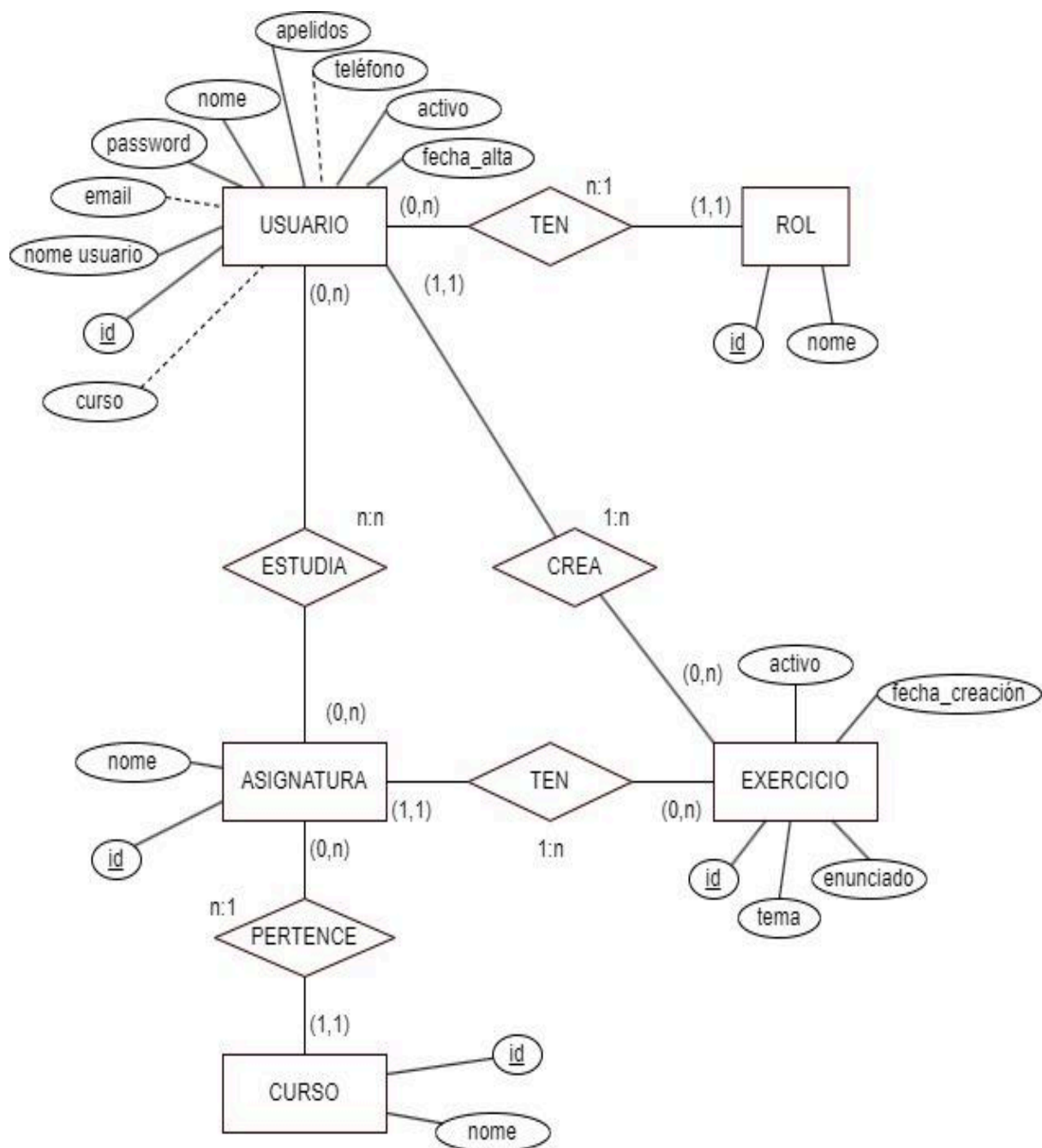
- Por un lado o **Modelo** que é a capa que representa a información coa que traballa o sistema e é a encargada de xestionar o acceso a dita información, tanto consultas como actualizacións.
- **Controlador:** encargado de xestionar eventos e comunicacións (normalmente solicitudes de usuarios) e invoca o Modelo cando se solicita a información que contén. Actúa como intermediario entre o Modelo e a Vista.
- **Vista:** encárgase de presentar a información do Modelo nun formato lexible adecuado para a interacción do usuario.

3.2.1. Xestor de bases de datos: MySQL

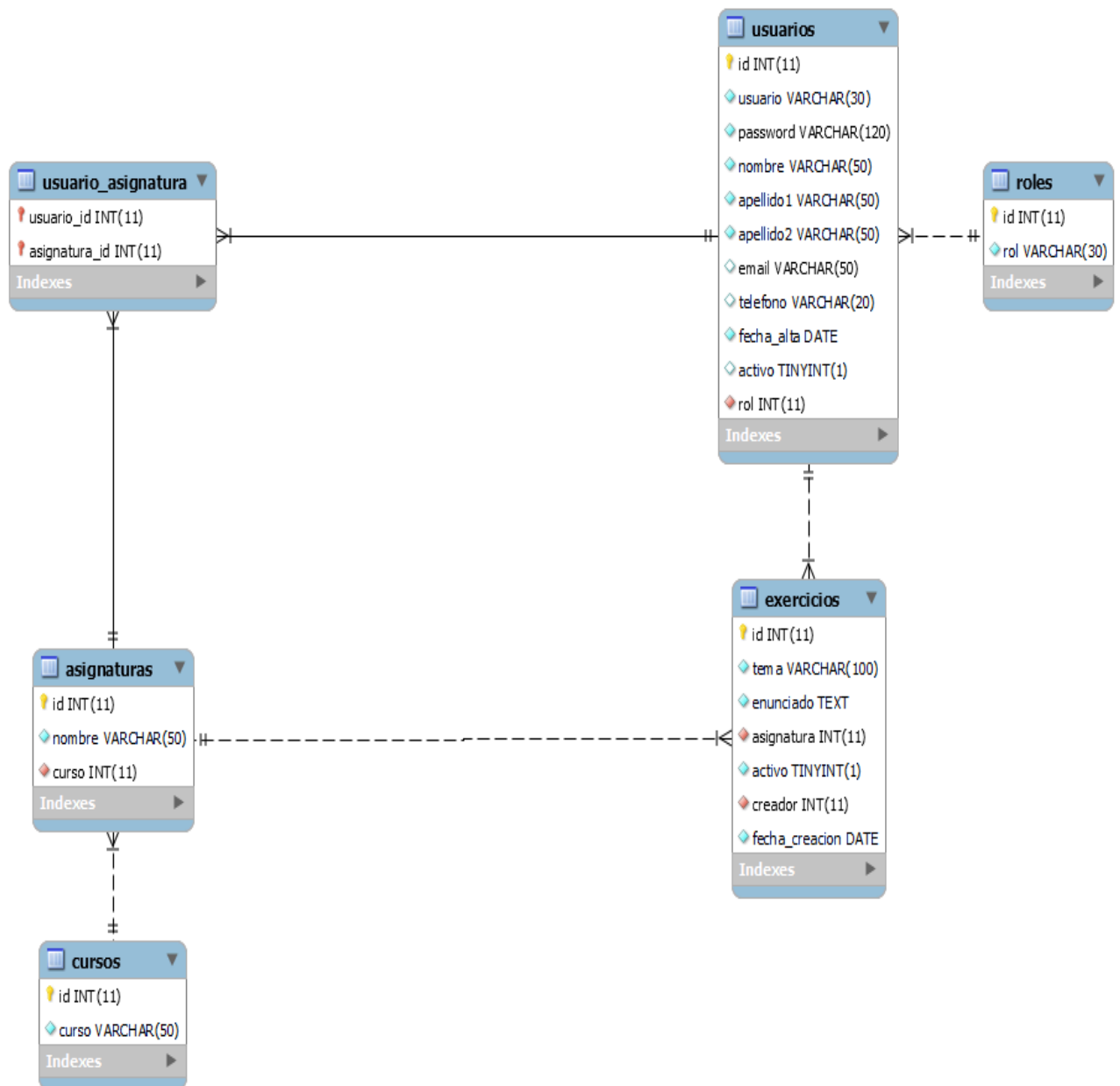
O primeiro que foi deseñado foi a base de datos, dado que é necesaria para calquera implementación que se desexe facer na aplicación. Para a súa creación e mantemento empregouse *MySQL*. Escollín este sistema xestor de bases de datos xa que é unha aposta segura, dado que é o *SGBD* mais usado na actualidade para proxectos web. É un sistema gratuito na súa versión *Community Edition*, fácil de usar, rápido, eficiente no consumo de recursos e é multiplataforma.

Para a administración da base de datos empregouse *phpMyAdmin*, unha ferramenta escrita en *PHP* que nos permite entre outras cousas crear e eliminar bases de datos, crear, eliminar e editar táboas, engadir, editar e eliminar campos, executar sentencias SQL etc. Todo isto dende unha cómoda interface web.

As seguinte imaxes correspóndense co Diagrama Entidade Relación e co esquema da base de datos respectivamente:



Imaxe 3.1: Diagrama Entidade Relación da base de datos



Imaxe 3.2: Esquema da base de datos

3.2.2. Editor de código: Visual Studio Code

Para o proceso de codificación empregouse *Visual Studio Code*, este é un editor de código fonte desenvolvido por *Microsoft*, gratuito e moi persoalizable grazas a unha gran cantidade de plugins que aumentan a súa funcionalidade. Ten soporte para depuración, integración con *Git*, resaltado de sintaxe e un largo etc que me fai pensar que esta é unha boa elección de cara a codificación do proxecto, ademais é un editor que xa coñezo e estou familiarizado con el logo de usalo o longo do curso e por isto último, sobre todo, foi polo que me decantei por el.

3.2.3. Linguaxes de programación: PHP e JavaScript

Como linguaxe de programación para a parte de servidor empregouse *PHP*. *PHP* é unha linguaxe de programación interpretada que se executa no servidor, é moi versátil, con unha boa documentación e conta con unha gran comunidade activa que lle dá un amplo soporte e actualización constantes.

Para a parte do cliente empregouse *JavaScript*, linguaxe de programación interpretada, orientada a obxectos e debilmente tipada, esta é executada polo navegador do usuario e é multiplataforma. Unha característica importante de JavaScript é que é unha linguaxe asíncrona o que nos permite executar procesos en segundo plano mentres o usuario interactúa con outros elementos da páxina.

Conta con unha gran variedade de librerías e *frameworks* e o igual que *PHP* ten unha comunidade moi activa e unha excelente documentación.

3.2.4. Framework de estilos da interface de usuario: Bootstrap

Bootstrap é un framework de código aberto que nos facilita o deseño e desenvolvemento de sitios web e aplicacións web ao proporcionar un conxunto de ferramentas, estilos predefinidos e compoñentes interactivos que nos permiten crear interfaces de usuario

modernas dun xeito sinxelo e eficiente. A diferenza de moitos frameworks web, só se ocupa do desenvolvemento front-end.

Algunhas das súas características son:

- **Reixa sensible:** *Bootstrap* usa un sistema de reixa fluída que se adapta dinamicamente ao tamaño da pantalla do dispositivo. Isto facilita a creación de deseños que se vexan ben nunha variedade de dispositivos, desde teléfonos móbiles ata pantallas de escritorio.
- **Compoñentes e estilos predefinidos:** *Bootstrap* inclúe unha gran variedade de compoñentes listos para usar, como botóns, formularios, navegación, táboas etc.
- **JavaScript incorporado:** *Bootstrap* inclúe complementos e scripts de *JavaScript* que melloran a funcionalidade dos compoñentes e permiten interaccións dinámicas na interface de usuario. Algúns exemplos inclúen fiestras modais, pestanas e controles deslizantes.
- **Compatibilidade entre navegadores:** foi deseñado para ser compatible cos principais navegadores web.
- **Documentación completa:** ten documentación detallada que ofrece instrucións claras sobre como usar cada compoñente e estilo.

3.2.5. Outras tecnoloxías empregadas

A programación da aplicación fíxose nun entorno de desenvolvemento local empregando *XAMPP* baixo un sistema operativo *Windows 10*.

XAMPP é un paquete de software libre que nos proporciona unha base de datos *MariaDB*, un servidor *Apache* e un intérprete para a linguaxe *PHP*, todo nun mesmo paquete e proporcionándonos unha fácil instalación e configuración.

Para o control de versión empregouse *Git* e *GitHub* como repositorio remoto. Outras tecnoloxías empregadas son as seguintes:

Composer

Composer é unha ferramenta de xestión de dependencias para a linguaxe de programación *PHP*. A súa función principal é simplificar e automatizar o proceso de xestión das bibliotecas e paquetes de software que precisa un proxecto *PHP* para funcionar correctamente.

Algunhas características principais de composer son:

- **Xestión de dependencias:** xestiona as dependencias do proxecto, o que facilita a instalación, actualización e eliminación de bibliotecas externas ou paquetes de software que a aplicación *PHP* precisa para funcionar.
- **Carga automática de clases:** facilita a carga automática de clases o que implica non ter que incluír manualmente cada ficheiro de clase, xa que *Composer* encárgase de cargar automaticamente as clases necesarias segundo sexa necesario.
- **Fácil instalación e actualización:** *Composer* simplifica a instalación e actualización de dependencias cun simple comando na liña de comandos. As dependencias descárganse do repositorio central de *Packagist*, un rexistro de paquetes para *PHP*.

Un dos ficheiros mais importantes de *Composer* é “composer.json”, onde se especifican as dependencias, versións e configuracións do proxecto. Este ficheiro tamén pode incluír información sobre scripts personalizados, configuración de carga automática e moito máis.

TinyMCE

TinyMCE é un editor de texto *WYSIWYG* (*What You See Is What You Get*) de código aberto para *HTML* que se executa enteiramente en *JavaScript* e distribúese libremente baixo a licenza *LGPL*.

O estar baseado en *JavaScript*, é independente da plataforma e execútase no navegador dos usuarios. Ten a capacidade de converter un campo de tipo *textarea* ou outros elementos *HTML* en instancias do editor, proporcionados un potente editor de texto.

As características principais de *TinyMCE* son:

- **Edición de texto enriquecido:** permite aos usuarios crear e editar contido visualmente, de forma similar a un procesador de textos, con opcións de formato como negriña, cursiva, listas e moito máis.
- **Persoalizable:** os desenvolvedores poden personalizar a aparencia e a funcionalidade do editor para adaptarse ás necesidades específicas da súa aplicación. Isto inclúe axustar as opcións da barra de ferramentas, engadir complementos e adaptar o estilo ao editor para que coincida co deseño xeral.
- **Compatibilidade entre navegadores:** está deseñado para funcionar perfectamente en varios navegadores web.
- **Soporte para complementos:** o editor admite un sistema de complementos que permite aos desenvolvedores ampliar a súa funcionalidade.
- **Filtrado de contido e seguridade:** inclúe mecanismos incorporados de filtrado de contido para mellorar a seguridade, evitando que se execute código potencialmente prexudicial.
- **Modos en liña e iframe:** os desenvolvedores poden escoller entre os modos en liña e iframe dependendo do estilo de integración desexado nas súas aplicacións web.

Adicionalmente completouse o editor con varios plugins, todos eles gratuítos, para intentar adaptalo as nosas necesidades. Un destes plugins de nome “*CodeCogs Equation Editor*” permítenos introducir nos enunciados dos exercicios todo tipo de ecuacións en formato *LaTeX*, que logo serán renderizadas a través da *API de CodeCogs* en imaxes que representan as ecuacións.

4. Instalación de proxecto

Unha vez o proxecto estaba nun estado avanzado decidín facer unha proba de implantación usando un servidor privado creado nunha instancia *Virtual Machine (VM) de Oracle Cloud Infrastructure (OCI)*. Esta empresa ofrece na súa nube, e de forma gratuíta, unha serie de recursos que podemos usar para crear instancias de máquinas virtuais.

Neste caso aproveitouse unha máquina virtual, xa creada con anterioridade, na que corre unha distribución *Ubuntu 22.04*, e instalouse todo o necesario para facer unha pila *LAMPP*, isto é:

- *MySQL* na versión 8.0.36
- servidor *Apache* na versión 2.4.52
- *PHP* na versión 8.1.2

Logo de configurar *Apache* e *MySQL* creouse a base de datos exportando os datos dende o *phpMyAdmin* local. Para crear a base de datos executamos na consola de comandos de *MySQL* o seguinte comando:

```
source academia.sql;
```

E logo creamos o usuario que usaremos para a conexión e asignámoslle os permisos necesarios

```
create user administrador@'localhost' identified by 'admin';
```

```
grant all on academia.* to administrador@'localhost';
```


Unha vez lista a base de datos creouse un *host virtual* no servidor apache co seguinte contido:

```
ubuntu@amperevm:~$ cat /etc/apache2/sites-enabled/ProxectoDAW.conf
<VirtualHost *:80>
    DocumentRoot /var/www/ProxectoDAW/public
    ServerName proxectodaw.duckdns.org
    ServerAlias www.proxectodaw.duckdns.org

    <Directory /var/www/ProxectoDAW/public>
        AllowOverride All
        options -Indexes
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
ubuntu@amperevm:~$
```

Imaxe 4.1: Virtual Host de Apache

Logo de crear o *host virtual* hai que activalo executando o seguinte comando

```
sudo a2ensite ProxectoDAW.conf
```

Unha vez está o servidor e a base de datos preparados clónase o repositorio de *GitHub*

```
git clone https://github.com/rikiotero/ProxectoDAW
```

Móvese a carpeta do proxecto a carpeta `/var/www/` de apache

```
sudo mv ProxectoDAW/ /var/www/ProxectoDAW/
```

Asignámoslle o grupo `www-data`

```
sudo chgrp www-data /var/www/ProxectoDAW
```

Asegurámosnos de que o noso usuario pertence o mesmo grupo

```
sudo usermod -a -G www-data $USER
```

Asignamos os permisos adecuados

```
sudo chmod -R 775 /var/www/ProxectoDAW
```

```
sudo chmod -R g+s /var/www/ProxectoDAW
```

Asegurámonos de ser os propietarios do directorio

```
sudo chown -R usuario /var/www/ProxectoDAW
```

E por último, situámonos no directorio */var/www/ProxectoDAW* e instalamos as dependencias necesarias con composer

```
composer install
```

Todo o anterior fíxose dende unha conexión *SSH* co servidor dende o cliente *PuTTY*.

O proxecto pode verse funcionando no seguinte enlace proxectodaw.duckdns.org. As credencias de acceso para probas foron as seguintes:

- Rol administrador:
 - Usuario: administrador
 - Contraseña: admin
- Rol profesor:
 - Usuario: profesor1
 - Contraseña: profesor1
- Rol estudante:
 - Usuario: estudiante1
 - Contraseña: estudiante1

Nota: para unha instalación baixo *Windows* é necesario mover as carpetas “*vendor/twbs/bootstrap/dist public/bootstrap*” e “*vendor/tinymce/tinymce/ public/tinymce*” os directorios “*public/bootstrap*” e “*public/tinymce*” respectivamente, xa que isto en Linux faíno un script declarado no “*composer.json*”.

5. Custos do proxecto

Os custos presentados a continuación son estimados xa que dado a actual oferta tanto de aluguer de hosting como de compra de dominio é realmente complicado facer unha estimación exacta sen saber os detalles do que a un cliente final lle interesaría.

Custos de infraestrutura

	Custo mes	Custo anual
Servidor	10€	120€
Dominio		10€
		Total: 130€ ano

Custos de desenvolvemento e mantemento

	Horas	Custo/hora	Total
Deseño	16	20€	320€
Codificación e despregue	70	20€	1400€
			Total: 1720€
Mantemento	2,5 hrs/mes	10€	300€ o ano

6. Melloras

Actualmente a aplicación inda que é funcional, de cara a unha versión final que sexa operativa nun entorno en produción teríamos que ter en conta varias cousas importantes, entre elas:

- Habería que xestionar dalgunha maneira a protección de datos dos usuarios, xa que traballando con datos persoais é unha obriga legal facelo.
- Faría falta engadir unha conta de “super administrador” que non se poida eliminar, dado que co sistema de rexistro actual é posible eliminar todos os usuarios.
- En canto o código, hai que mellorar o tratamento das excepcións e sacar as mensaxes de erro por pantalla.

En canto a melloras que engadirían funcionalidade a aplicación pero que non serían fundamentais a curto prazo para poder por en produción a aplicación:

- A xeración de contrasinais aleatorios, seguindo certas restricións, e que foran enviadas por mail os usuarios unha vez son creados, evitando así que o administrador teña que inventarse unha contrasinal por cada usuario que rexistre.
- Un mellor sistema de cambio ou recuperación de contrasinal, usando o correo electrónico do usuario para facelo.
- Permitir engadir imaxes e vídeos nos enunciados dos exercicios, ou asociados o exercicio dalgunha maneira.
- Poder almacenar tamén temarios das materias e exámenes, podendo organizar, por exemplo, exámenes de EBAU por ano e materia.
- Permitir ordear todas as listas segundo desexemos.
- Mellorar o sistema de notificacións.
- Mellorar visualmente a interface de usuario.

7. Manual de usuario

Para o acceso a aplicación é preciso ter unha conta de usuario. Existen tres tipos de contas de usuario:

- Administrador
- Profesor
- Estudiante

As contas de usuario só poden ser dadas de alta por un usuario “administrador”, ou por un usuario “profesor” no caso dos estudantes. As contas de usuario poden estar activadas ou desactivadas, polo que para o acceso a aplicación é preciso ter unha conta rexistrada e activada.

Logo de ter unha conta coa que iniciar sesión, e segundo sexa o tipo da conta, podemos acceder a seguinte pantalla:

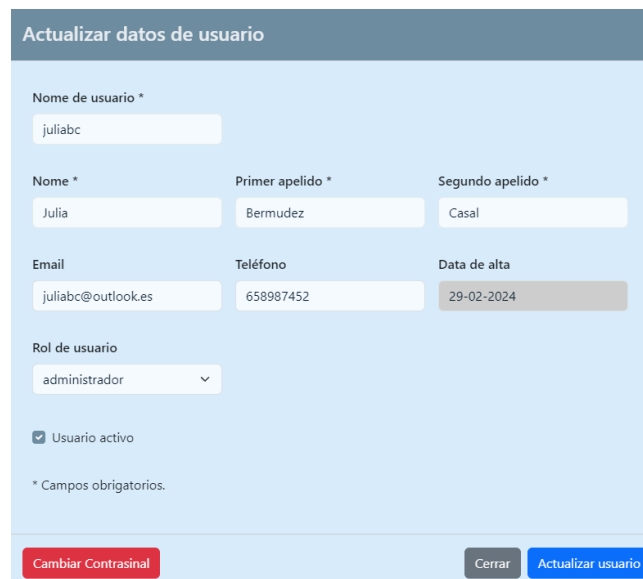
7.1. Panel de administrador

O panel de administrador conta coas seguintes pestanas:

- **Datos de usuario:** mostra os datos da conta de usuario e permite actualizalos.
- **Xestión de usuarios:** mostra a lista de todos os usuarios rexistrados no sistema e permite dar de alta novos usuarios, eliminalos e actualizar os seus datos.
- **Xestión de cursos e materias:** mostra a lista de todos os cursos e materias rexistrados no sistema, e permite crear novos cursos e materias, eliminar ou actualizar os xa existentes.

7.1.1. Pestana datos de usuario

A pestana de “datos de usuario” mostra os datos da conta. O pulsar o botón de “Actualizar datos” ábrese unha ventana modal que permite actualizar tanto os datos de usuario como cambiar o contrasinal da conta.



The screenshot shows a modal window titled "Actualizar datos de usuario". It contains several input fields: "Nome de usuario *" with the value "juliabc", "Nome *" with "Julia", "Primer apellido *" with "Bermudez", and "Segundo apellido *" with "Casal". Below these are "Email" (juliabc@outlook.es), "Teléfono" (658987452), and "Data de alta" (29-02-2024). There is a dropdown for "Rol de usuario" set to "administrador" and a checked checkbox for "Usuario activo". A note at the bottom states "* Campos obligatorios.". At the bottom of the modal are three buttons: "Cambiar Contraseña" (red), "Cerrar" (grey), and "Actualizar usuario" (blue).

Imaxe 7.1: Ventana modal de actualizar datos de usuario

7.1.2. Pestana xestión de usuarios


A pestana de “xestión de usuarios” mostra unha lista de todos os usuarios rexistrados, permite filtrar a lista de usuarios segundo uns certos criterios como o rol de usuario ou o estado activo da conta. O cadro de busca permite buscar usuarios filtrando por calquera das columnas que mostra a lista.



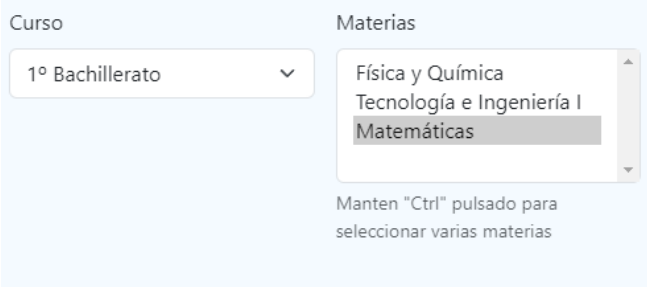
The screenshot shows the filter section of the user management interface. It includes checkboxes for "Usuarios activos" and "Usuarios inactivos", a dropdown for "Roles...", and a search input field labeled "Buscar". A red arrow points to the "Buscar" field. Below the filters, a red text box contains the text: "Permite filtrar por calquera columna da táboa de usuarios".

Imaxe 7.2: Filtros da lista de usuarios



A columna de accións conta con dous botóns para xestionar os usuarios, o primeiro é para editar o usuario e o segundo  para eliminar a conta.

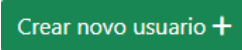
O pulsar o botón de editar usuario ábrese unha ventana modal que permite modificar os datos persoais do usuario, o seu rol e o estado da conta “activada/desactivada” (as contas de usuario desactivadas non poderán iniciar sesión). No caso de ser un usuario estudante tamén permite asignarlle un curso e materia, a asignación de varias materias realizase mantendo pulsado “Ctrl” e pinchando nas que se desexe seleccionar. Desde esta ventana tamén temos acceso o cambio de contrasinal da conta de usuario.



The image shows a modal window for editing a user. It has two main sections: 'Curso' and 'Materias'. The 'Curso' section has a dropdown menu currently showing '1º Bachillerato'. The 'Materias' section has a list box containing 'Física y Química', 'Tecnología e Ingeniería I', and 'Matemáticas', with 'Matemáticas' currently selected. Below the list box, there is a text instruction: 'Manten "Ctrl" pulsado para seleccionar varias materias'.

Imaxe 7.3: Selección de curso e materias na ventana modal de editar usuario

Crear usuarios

Nesta pestana tamén se pode dar de alta novos usuarios pulsando o botón . Logo de pulsalo ábrese unha ventana modal cun formulario para cubrir cos datos do novo usuario. É importante prestar atención a selección do rol do usuario que se está rexistrando, por defecto a aplicación selecciona o rol de estudante. Para a selección de múltiples materias, de ser o caso, realizase mantendo pulsado “Ctrl” e pinchando nas materias que se desexe seleccionar.

The image shows a modal form titled "Rexistro de novo usuario" with a light blue background. It contains several input fields and dropdown menus arranged in a grid. The fields are: "Nome de usuario *" (text), "Contrasinal *" (text), "Repetir contrasinal *" (text), "Nome *" (text), "Primer apelido *" (text), "Segundo apelido *" (text), "Email" (text, with example "Ex: email@info.gal"), "Teléfono" (text, with example "Ex: 666999333"), "Data de alta" (date, with value "14-03-2024"), "Rol de usuario *" (dropdown, with value "estudiante"), "Curso *" (dropdown, with value "Selección de curso"), and "Materias *" (multi-select dropdown, with value "Selección de materias"). Below the "Materias" dropdown is a note: "Manten 'Ctrl' pulsado para seleccionar varias materias". At the bottom left, there is a checkbox labeled "Usuario activo" which is checked, and a note "* Campos obligatorios.". At the bottom right, there are two buttons: "Cerrar" (grey) and "Rexistrar usuario" (blue).

Imaxe 7.4: Ventana modal de rexistro de usuario

7.1.3. Pestana xestión de cursos e materias

A pestana de “xestión de cursos e materias” mostra unha lista dos cursos e materias rexistradas, estas poderán ser asignadas os estudantes segundo son dados de alta ou logo editando o usuario dende a pestana de “xestión de usuarios”.

Para rexistrar un curso novo basta con engadir o nome do curso e pulsar o botón de “Engadir curso”.

The image shows a form with a light blue background. It has a text input field with the placeholder "Escribe o nome do curso novo" and a green button labeled "Engadir curso +". Below the input field, there is a note: "Para insertar un curso novo escribe o nome do curso e pulsa en 'Engadir curso'".

Imaxe 7.5: Engadir curso

Dende a lista de cursos pódese editar o nome do curso ou eliminalo pulsando nos botóns de “editar” ou “borrar” respectivamente. Para poder borrar un curso este non pode ter materias asociadas, de ser o caso é preciso eliminar primeiro todas as materias do curso.



7	4º ESO		
8	1º Bachillerato		
9	2º Bachillerato		

Imaxe 7.6: Edición ou borrado do curso

Do mesmo xeito se poden engadir materias os cursos xa rexistrados, para isto primeiro é preciso seleccionar o curso no que se quere engadir a materia, logo basta con escribir o nome da materia e pulsar o botón de “Engadir materia”.



1º seleccionar curso 4º ESO

2º nome da materia **Engadir materia o curso +**

Para insertar unha materia nova selecciona primeiro o curso donde queres insertala

lista de materias do curso seleccionado

Materias		
Id	Materia	Accións
4	Tecnología	 
8	Matemáticas	 
12	Biología y Geología	 

Imaxe 7.7: Creación dunha materia

Unha vez seleccionado un curso móstranse unha lista de materias asociadas a dito curso, dende esta lista pódese editar o nome da materia ou borrarla, pulsando nos respectivos botóns. Para poder eliminar unha materia esta non pode ter exercicios asociados.

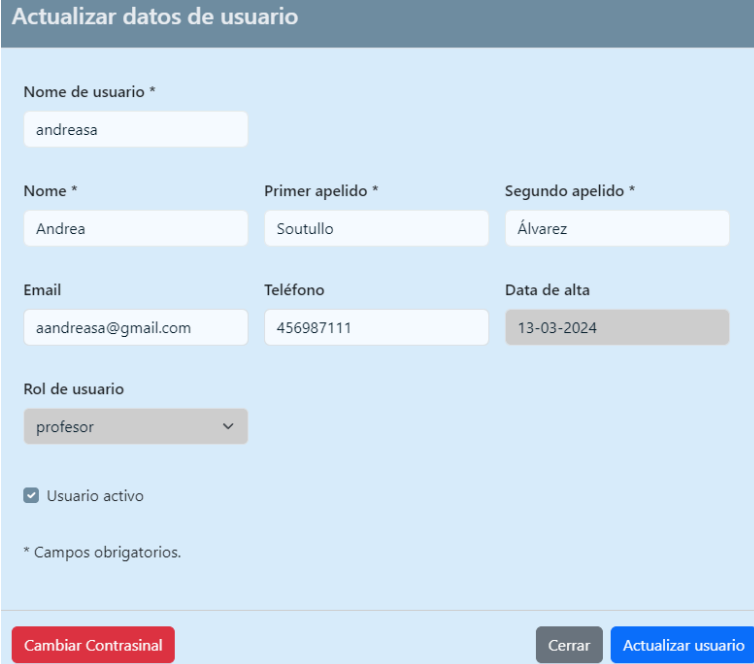
7.2. Panel de profesor

O panel de profesor conta coas seguintes pestanas:

- **Datos de usuario:** mostra os datos da conta de usuario e permite actualizalos.
- **Xestión de alumnos:** mostra a lista de todos os alumnos rexistrados no sistema e permite dar de alta novos alumnos, eliminalos e actualizar os datos.
- **Xestión de exercicios:** mostra unha lista con todos os exercicios creados e permite crear novos exercicios, eliminalos e actualizalos.

7.2.1. Pestana datos de usuario

A pestana de “datos de usuario” mostra os datos da conta. O pulsar o botón de “Actualizar datos” ábrese unha ventana modal que permite actualizar tanto os datos de usuario como cambiar o contrasinal da conta.



Actualizar datos de usuario

Nome de usuario *

andrea

Nome * Primer apellido * Segundo apellido *

Andrea Soutullo Álvarez

Email Teléfono Data de alta

aandrea@gmail.com 456987111 13-03-2024

Rol de usuario

profesor

☒ Usuario activo

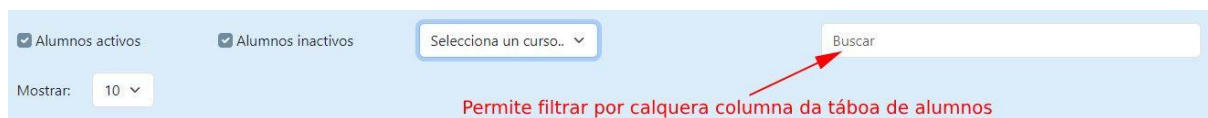
* Campos obligatorios.

Cambiar Contraseña **Cerrar** **Actualizar usuario**

Imaxe 7.8: Ventana modal de actualizar datos de usuario

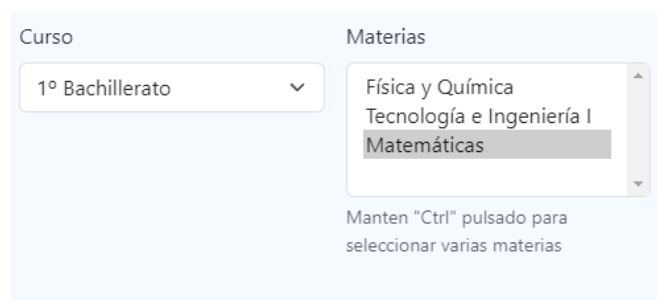
7.2.2. Pestana xestión de alumnos

A pestana de “xestión de alumnos” mostra unha lista de todos os alumnos rexistrados, permite filtrar a lista de alumnos segundo uns certos criterios como o curso ou o estado activo da conta. O cadro de busca permite buscar alumnos filtrando por calquera das columnas que mostra a lista.



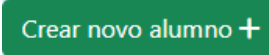
Imaxe 7.9: Filtros da lista de alumnos

O pulsar o botón de editar alumno ábrese unha ventana modal que permite modificar os datos persoais do alumno e o estado da conta “activada/desactivada” (as contas de alumno desactivadas non poderán iniciar sesión). Tamén permite asignarlle un curso e materia, a asignación de varias materias realizase mantendo pulsado “Ctrl” e pinchando nas que se desexe seleccionar. Desde esta ventana tamén temos acceso o cambio de contrasinal da conta de usuario.

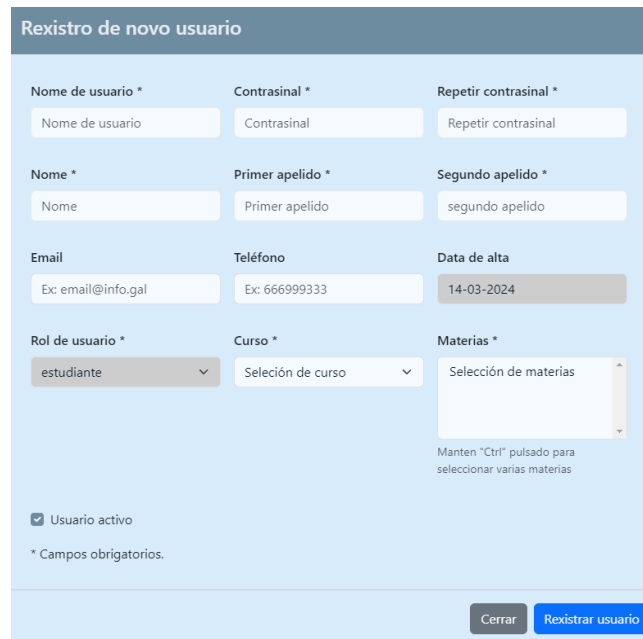


Imaxe 7.10: Selección de curso e materias na ventana modal de editar alumnos

Crear alumnos

Nesta pestana tamén se pode dar de alta novos alumnos pulsando o botón . Logo de pulsalo ábrese unha ventana modal cun formulario para cubrir cos datos do novo

alumno. Para a selección de múltiples materias, de ser o caso, realizase mantendo pulsado “Ctrl” e pinchando nas materias que se desexe seleccionar.



Formulario de registro de novo usuario. O formulario está dividido en varias seccións:


- Nome de usuario ***: Campo de texto para o nome de usuario.
- Contrasinal ***: Campo de texto para a contrasinal.
- Repetir contrasinal ***: Campo de texto para repetir a contrasinal.
- Nome ***: Campo de texto para o nome.
- Primer apelido ***: Campo de texto para o primeiro apelido.
- Segundo apelido ***: Campo de texto para o segundo apelido.
- Email**: Campo de texto para o email, co exemplo "Ex: email@info.gal".
- Teléfono**: Campo de texto para o teléfono, co exemplo "Ex: 666999333".
- Data de alta**: Campo de texto para a data de alta, co exemplo "14-03-2024".
- Rol de usuario ***: Selector de rol, co valor predeterminado "estudiante".
- Curso ***: Selector de curso, co valor predeterminado "Selección de curso".
- Materias ***: Selector de materias, co valor predeterminado "Selección de materias".

Debaxo dos campos, hai un checkbox "Usuario activo" e un asterisco explicando que os campos son obrigatorios. Na parte inferior dereita, hai botóns "Cerrar" e "Registrar usuario".

Imaxe 7.11: Ventana modal de rexistro de alumno

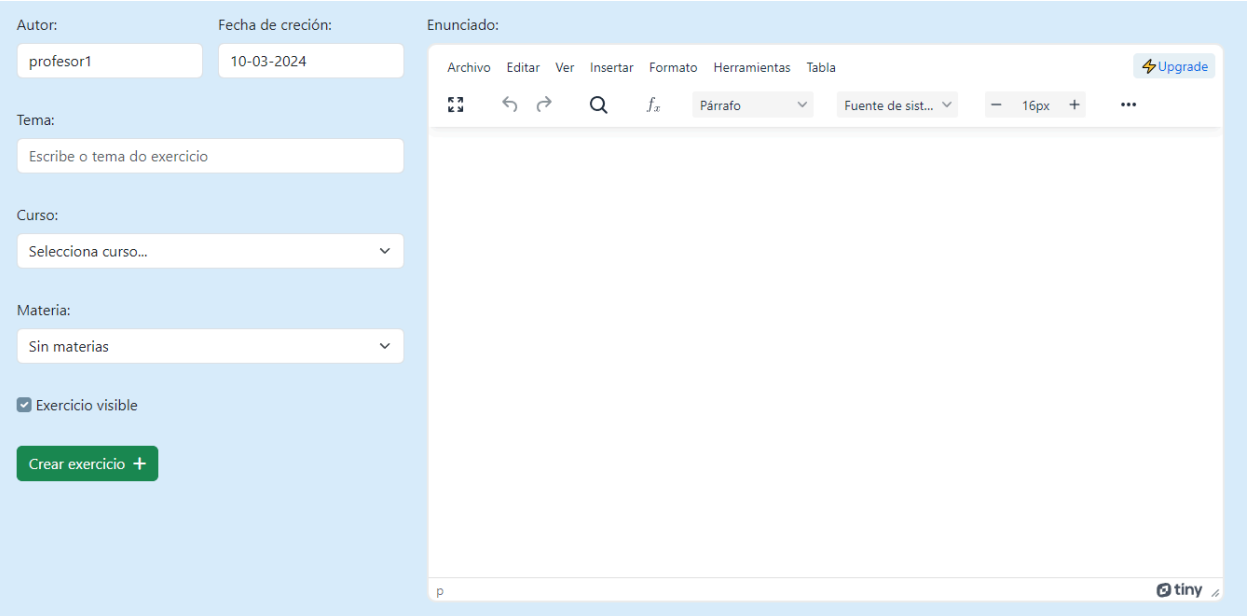
7.2.3. Pestana de xestión de exercicios

A pestana de “xestión de exercicios” mostra unha lista de todos os exercicios rexistrados, permite filtrar a lista de exercicios segundo certos criterios. O cadro de busca permite buscar exercicios filtrando por calquera das columnas que mostra a lista e tamén permite facer buscas polo enunciado do exercicio.

A columna de accións conta con dous botóns para xestionar os exercicios, o primeiro abre o editor de exercicios, o segundo  é para eliminalo.

Crear ejercicios

Nesta pestana tamén se pode crear novos exercicios pulsando o botón **Crear ejercicio +** logo de pulsalo ábrese o editor de exercicios. Neste hai que cubrir os campos “Tema”, “Curso” e “Materia” e seleccionar si o exercicios estarán visible ou non para os alumnos do curso seleccionado.

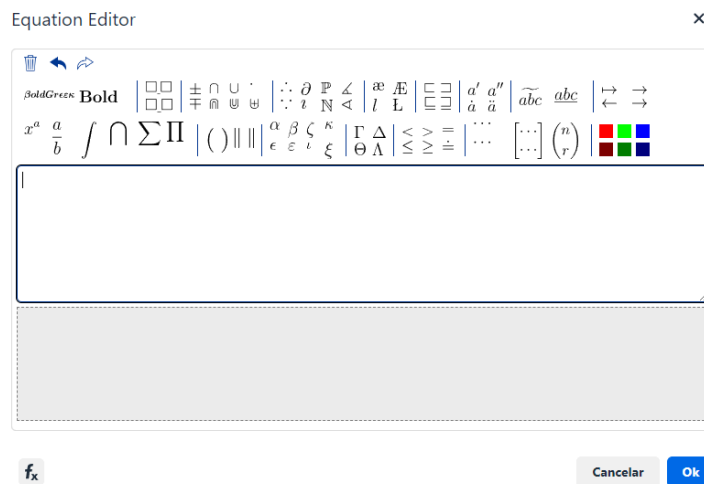


Imaxe 7.15: Editor de exercicios


Editor de enunciados

O editor de enunciados é un editor de texto enriquecido que entre outras características conta con selección de tipo e tamaño de fonte, busca de texto, inserción de táboas, negriña, cursiva etc.

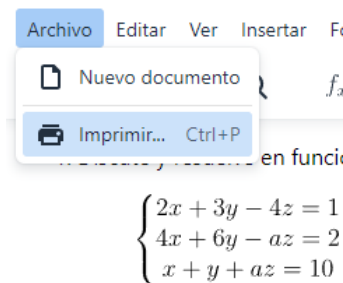
O botón f_x abre un editor de fórmulas que permite la inserción de todo tipo de fórmulas matemáticas.



Imaxe 7.16: Editor de fórmulas

No botón  pódese alternar a pantalla completa para maior comodidade.

Dende o menú Archivo-Imprimir pódese tanto imprimir como descargar o enunciado en formato PDF.



Imaxe 7.17: Impresión ou descarga dun enunciado

7.3. Panel de estudante

O panel de estudante conta coas seguintes pestanas:

- **Datos de usuario:** mostra os datos da conta de usuario e permite actualizar as credenciais de acceso, o email e o teléfono.

- **Exercicios:** mostra unha lista con todos os exercicios que hai activos para as materias do alumno.

7.3.1. Pestana datos de usuario

A pestana de “datos de usuario” mostra os datos da conta. O pulsar o botón de “Actualizar datos” ábrese unha ventana modal que permite actualizar o nome de usuario, email, teléfono e o contrasinal da conta.

Nome de usuario *

estudiante1

Nome *

Juan

Primer apellido *

Caamaño

Segundo apellido *

Del Toro

Email

caamanodeltoro@terra.com

Teléfono

654349876

Data de alta

30-01-2024

Rol de usuario

estudiante

Curso *

1º Bachillerato

Materias *

Física y Química
Tecnología e Ingeniería I
Matemáticas

Manten "Ctrl" pulsado para seleccionar varias materias

* Campos obligatorios.

Cambiar Contraseña


Cerrar

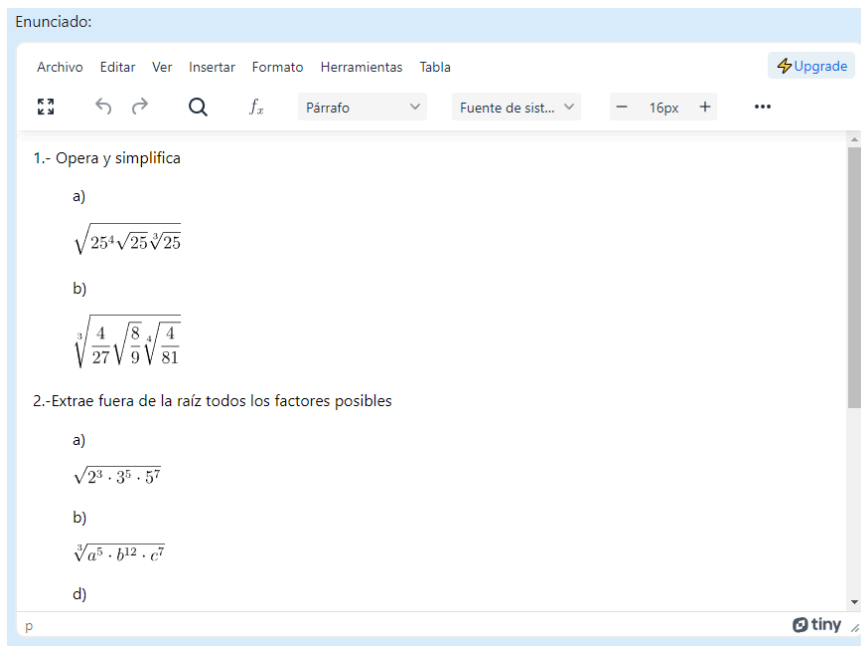
Actualizar usuario

Imaxe 7.18: Ventana modal de actualizar datos de usuario


7.3.2. Pestana de exercicios

A pestana de “exercicios” mostra unha lista de todos os exercicios activos para as materias do alumno, permite filtrar a lista de exercicios segundo certos criterios. O cadro de busca permite buscar exercicios filtrando por calquera das columnas que mostra a lista e tamén permite facer buscas polo enunciado do exercicio.

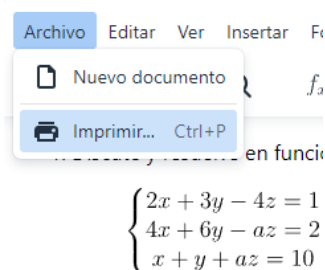
O pulsar o botón “ver exercicio”  ábrese unha nova ventana onde o estudante pode consultar o enunciado do exercicio.



Imaxe 7.19: Vista do enunciado dun exercicio

No botón  pódese alternar a pantalla completa para maior comodidade.

Dende o menú Archivo-Imprimir pódese tanto imprimir como descargar o enunciado en formato PDF.



Imaxe 7.20: Impresión ou descarga dun enunciado

8. Bibliografía

- <https://www.php.net/manual/es/index.php>
- <https://diego.com.es/certificacion-php>
- <https://www.w3schools.com/>
- <https://developer.mozilla.org/es/>
- <https://app.diagrams.net/>
- <https://www.tiny.cloud/docs/tinymce/6/>
- https://editor.codecogs.com/docs/3-Tiny_MCE_v6.php
- <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- <https://www.oracle.com/cloud/free/>