PROJECTE FINAL

**Logotipo

Descripción generada automáticamente**

**Projecte presentat per:**

*Martín Hernan Jaime Bonvin*

**Estudiant de:**

2n Grau de Desenvolupament d’Aplicacions Web

# INDEX

[INDEX 2](#_Toc167829348)

[Introducció: 3](#_Toc167829349)

[Requeriments funcionals 4](#_Toc167829350)

[- Principis: 4](#_Toc167829351)

[- Projecte de disseny web: 4](#_Toc167829352)

[- Menús d’un Projecte: 5](#_Toc167829353)

[- Configuració usuari: 7](#_Toc167829354)

[Requeriments tècnics 8](#_Toc167829355)

[Front End: 8](#_Toc167829356)

[Back End: 8](#_Toc167829357)

[Diagrama cas d’ús 10](#_Toc167829358)

[- Usuari anònim: 10](#_Toc167829359)

[- Usuari registrat: 10](#_Toc167829360)

[Diagrama d’Activitats 10](#_Toc167829361)

[- Usuari anònim: 10](#_Toc167829362)

[- Usuari registrat: 12](#_Toc167829363)

[Interfície d’usuari 14](#_Toc167829364)

[Wireframe usuari anònim: 14](#_Toc167829365)

[Wireframe usuari registrat: 19](#_Toc167829366)

[Mockup usuari anònim: 25](#_Toc167829367)

[Mockup usuari registrat: 29](#_Toc167829368)

[Diagrama Base de dades: 35](#_Toc167829369)

[- Usuaris: 35](#_Toc167829370)

[- Projectes: 36](#_Toc167829371)

[Implementació: 38](#_Toc167829372)

[Front End: 38](#_Toc167829373)

[Back End: 39](#_Toc167829374)

[Planificació temporal 41](#_Toc167829375)

# Introducció:

La idea principal d'aquest projecte és crear una **interfície de gestió de disseny de pàgina web** per a l'usuari basat en la funcionalitat "***What you see is what you get***", de manera que sigui senzilla, fàcil i intuïtiva d’utilitzar. La funcionalitat d'aquesta és dissenyar una web (no real) per tenir una idea clara sobre com es vol desenvolupar una pàgina web, és a dir, poder crear un "***Mockup***" o un "***Wireframe***" a partir d'elements web.

Aquesta conté una funcionalitat d'elements basada en [**Elementor**](https://elementor.com) (plugin de WordPress) i [**WordPress**](https://wordpress.com), on la seva funcionalitat es basa en una manera fàcil de desenvolupar una pàgina web a partir d’arrossegar elements a contenidors per ubicar l’element desitjat a una zona especifica de la pàgina web.

Aquest funciona a partir de codi JavaScript i elements HTML amb l'extensió de [Bootstrap](https://getbootstrap.com), és a dir, cada element que s'apliqui al disseny d'un projecte web, passarà per un procés de creació d'elements HTML amb JavaScript perquè hi hagi la visualització desitjada.

Aquesta pàgina web de gestió web permet a l'usuari registrar-se i iniciar sessió perquè el progrés de desenvolupament web es **guardi a la base de dades**, a més de tenir la configuració bàsica d'usuari com editar el compte d'usuari canviant el **nom d'usuari**, la **contrasenya**, o **esborrar el compte** o els **projectes** **ja creats**.

# Requeriments funcionals

### Principis:

La funcionalitat principal de la pàgina web és tenir la possibilitat de crear diversos projectes de disseny web, on serà necessari **registrar-se** per que es puguin guardar a la base de dades.

En cas que un usuari **no estigui registrat**, podrà crear un projecte i editar-lo, però **no** tindrà l'opció de guardar aquest en cap compte.

En la pàgina principal existirà la visualització de tots els projectes creats per l'usuari que hagi iniciat sessió. En cada requadre de projecte hi haurà un desplegable amb les opcions: **esborrar el projecte**, **canviar el nom del projecte**, **duplicar el projecte**, o **visualitzar la informació del projecte**. En aquest últim cas, la informació a mostrar a partir d’un pop-up serà **l'identificador del projecte**, **la** **data i hora en què va ser creat** i **el** **nom del projecte**.

### Projecte de disseny web:

L'usuari podrà seleccionar una de les **plantilles base** creades per defecte que es mostraran abans de començar a afegir elements. En cas que l'usuari ja tingui un arxiu JSON on poder començar a treballar, podrà **importar** l'arxiu i afegir elements a partir del funcionament de la pàgina. En cada projecte es treballa sobre un codi HTML i en el procés de desserialitzar el contingut a JSON i viceversa perquè es pugui emmagatzemar a la base de dades i tornar a mostrar el projecte a la pàgina.

Un cop l'usuari hagi creat un projecte, aquest contindrà els elements base per crear una pàgina (**Encapçalat, cos, i peu de pàgina**).

A partir d'aquí existiran **containers** amb les opcions d'editar **elements** (Títol, imatge, text, llistes, etc.) dins d'aquests, crear nous containers, moure'ls de lloc, entre altres.

En el següent **exemple** es pot veure tres containers dividits en tres fileres:

* La primera fila conté un objecte de **títol** en la primera columna. Conté una segona columna buida i un text per defecte en la tercera columna.
* En la segona fila, el segon container conté 4 columnes buides (D’aquesta manera vull donar a entendre que es podrà seleccionar containers amb un nombre de columnes).
* I una tercera fila amb un container buit.

***Demostració****:*

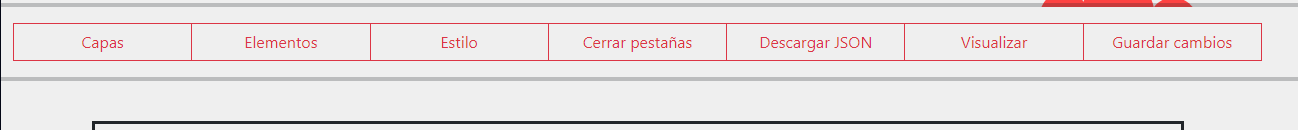
Gráfico

Descripción generada automáticamente con confianza baja

### Menús d’un Projecte:

Cada projecte conté un **menú superior** amb les opcions de poder visualitzar les diferents barres flotants existents. La funcionalitat d’aquest menú, és la de **mostrar o tancar** aquestes barres per una millor visualització del projecte i que no arribi a ser tan "carregant" o "pesat" treballar en ell.

***Demostració****:*



Cada menú conté una funcionalitat especial pels elements seleccionats o pels que es vol crear, aquestes ajuden a tenir major consciència de la configuració del projecte, de cada element aplicat i de quins nous elements es vol aplicar.

Cada element seleccionat conté una barra de configuració per **l'estil** d'aquest, és a dir, poder **editar la mida**, **aplicar colors**, **tipografia**, etc.

***Demostració****:*

Imagen que contiene Forma

Descripción generada automáticamente

A més, es té en compte una barra flotant per visualitzar i editar les **capes** de cada element. La idea d'aquesta és tenir major coneixement des d'on s'està editant cada element.

***Demostració****:*

Imagen que contiene Gráfico

Descripción generada automáticamente

Hi ha una barra flotant per poder visualitzar **els elements predeterminats** existents a voler aplicar. Es a partir d'aquí que l'usuari pot aplicar arrossegant directament els elements desitjats a visualitzar a cada container.

***Demostració****:*

Gráfico

Descripción generada automáticamente con confianza media

A més, l'usuari té l'opció de **descarregar** tot el projecte com un arxiu JSON, perquè el pugui **editar o visualitzar** de manera local el contingut de tot el projecte i tornar a importar aquest en un projecte nou de la mateixa pàgina.

En cas que l'usuari vulgui previsualitzar el seu projecte com una pàgina web, l'opció de **visualització** li permet veure’l en una pestanya a part.

### Configuració usuari:

L'usuari té l'opció de **modificar** el seu compte canviant el **nom d'usuari**, enviant-lo a un formulari bàsic per escollir el seu nou nom.

Pot **canviar la contrasenya** a partir d’un formulari per aplicar la contrasenya antiga i la nova dues vegades amb les restriccions necessàries per que hi hagi major seguretat..

També podrà **esborrar** el seu compte d'usuari juntament amb els projectes guardats al compte.

En cas que l'usuari hagi **oblidat la contrasenya**, existeix l'opció de canviar la contrasenya a partir d'un formulari on s'haurà d'aplicar el correu electrònic, aquest enviarà al seu correu electrònic el link a un formulari per poder aplicar la nova contrasenya dues vegades amb les restriccions necessàries per a una major seguretat. Aquest també funciona a partir de canviar el **token** de l'usuari emmagatzemat a la base de dades i comprovar si és el mateix que hi conté el link de canviar la contrasenya per a una major seguretat.

# Requeriments tècnics

Els **requeriments tècnics** pel correcte funcionament de la pàgina web han sigut:

## Front End:

* **JavaScript**: La idea principal del projecte és el funcionament de creació d’elements DOM i l’aplicació d’aquests per la visualització del disseny web. Per això, he escollit el llenguatge de programació **JavaScript** per la correcta gestió d’aquest.

A més he fet servir diversos **APIs** amb el propòsit de facilitar la gestió d’actualitzar el projecte a la **base de dades** i el funcionament d’actualitzar els **elements DOM**. Aquests són:

* + **JQuery:** El funcionament d’aquest API m’ha permès tenir una major gestió d’elements de manera que pugui tenir un codi més **simplificat**.
  + **Fetch:** Amb la llibreria **Fetch** he realitzat la connexió als arxius PHP per **l’actualització de dades dels usuaris i dels projectes**.
  + **MutationObserver:** Aquesta API em permet detectar cada **actualització de contingut** en el projecte. La seva funció principal és detectar quan el body del projecte està **buit** i és necessari aplicar el **botó de creació d’un nou container**.
* **Bootstrap 5.1:** La llibreria **Bootstrap** ofereix una manera fàcil de fer servir **l’estil de cada element** i una **visualització més atractiva** d’aquestes. Aquesta llibreria s’ha fet servir tant en l’edició de projectes, com en la pàgina **Weizz** en general.
* **JSON:** El llenguatge de programació **JSON** permet **simplificar a text** tot el contingut d’un projecte i que sigui emmagatzemat a la base de dades. És des d’aquest llenguatge que es realitza l’acció de **desserialitzar** contínuament els projectes per poder importar aquests i que es guardin a la base de dades.

## Back End:

* **PHP 8.1:** Per la creació de les pàgines amb codi HTML i el correcte funcionament de peticions web y construcció d’informació com ara la visualització dels projectes disponibles per a cada usuari, emmagatzemar les dades de cada projecte editat, emmagatzemar les dades de l’usuari registrat, entre altres.

He decidit crear la funcionalitat tècnica amb **PHP natiu,** ja que no he considerat que hagi necessitat llibreries per les funcions d’actualitzar les dades d’un usuari i projecte o creació d’usuaris i projectes.

Per el correcte funcionament de gestió d’actualització de dades a la base de dades he fet servir les següents extensions:

* + **PDO:** L’extensió PDO facilita la connexió a la base de dades, i permet crear consultes i modificacions a partir de codi **MySQL.**
* **PHPMailer:** Aquesta llibreria de **PHP** em permet fer una connexió a un servidor **SMTP** per enviar un missatge al **correu electrònic desitjat**. Aquesta extensió s’ha fet servir principalment per enviar un formulari de **modificació de contrasenya** al correu aplicat en cas que l’usuari **no recordi** la seva contrasenya d’usuari.
* **XAMPP:** He fet servir XAMPP per crear una instancia **local** i comprovar el correcte funcionament de la meva aplicació web, a més de poder crear una base de dades en local amb **phpMyAdmin** on s’emmagatzema totes les dades**.**
* **Hestia:** Hestia és l’aplicació de **panell de control del servidor web** que he fet servir per emmagatzemar els arxius del projecte i desplegar l’aplicació web. Aquest conté una interfície fàcil de fer servir per a una correcta gestió de dominis i de pàgines, a més compte amb **phpMyAdmin** amb **MariaDB** per l’emmagatzemat de dades a una base de dades.

# Diagrama cas d’ús

### Usuari anònim:

Un usuari anònim només té l'opció de **registrar-se** o **iniciar sessió** en cas que ja tingui un compte. A més, només podrà crear un projecte simultàniament **sense l’opció** de guardar el projecte. Podrà descarregar el contingut JSON del projecte i importar aquest com un altre projecte.

A més, en cas que un usuari hagi oblidat la contrasenya d'usuari, tindrà l'opció d'omplir un formulari on únicament serà necessari aplicar el correu electrònic per i que s'enviï **l'enllaç de recuperació de contrasenya**, aquest obrirà un nou formulari amb els camps per aplicar una **nova contrasenya** al correu electrònic seleccionat.

### Usuari registrat:

Un usuari ja registrat, sí que tindrà habilitades les opcions de guardar diferents projectes simultàniament.

També tindrà totes les configuracions d'usuari. Aquestes són canviar el nom d'usuari i la contrasenya. També hi tindrà l'opció d'eliminar permanentment el seu compte i els projectes creats en ell.

***Diagrama complet***:

Imagen que contiene Diagrama

Descripción generada automáticamente

# Diagrama d’Activitats

### Usuari anònim:

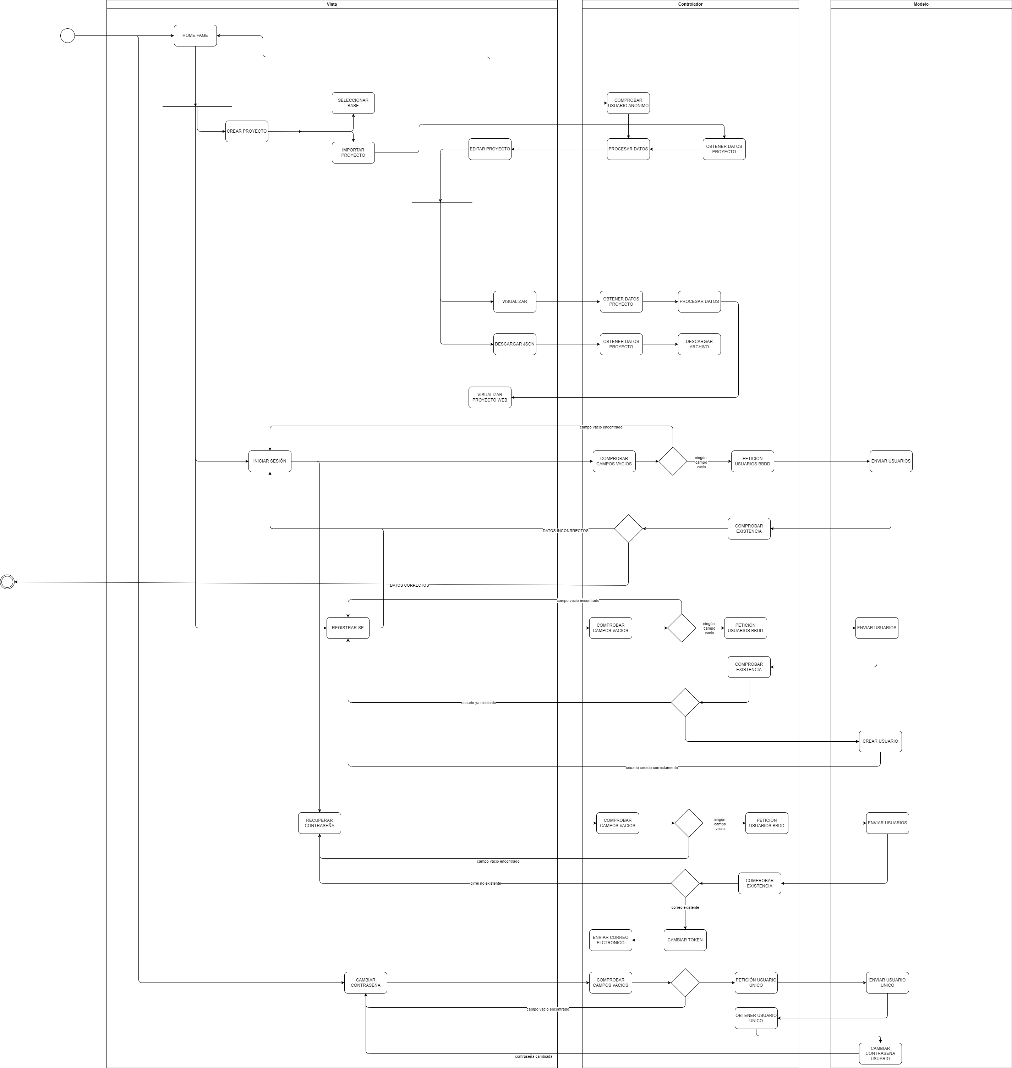
A l'hora de crear un projecte en el cas d'un **usuari anònim**, les dades per les bases dels projectes s'agafa directament des d'uns **arxius JSON** emmagatzemats en el servidor. El programa obté aquests arxius i realitza la **desserialització** per mostrar la **base principal** seleccionada.

A partir d'aquí l'usuari podrà editar el projecte amb les opcions **d'edició complet**, i amb l'opció de **visualitzar** i **descarregar** el projecte com a extensió JSON.

En els formularis d'iniciar sessió i registrar-se, el codi realitza totes les comprovacions necessàries per saber si s'han complert tots els **requisits d'autenticació**, és a dir, que les dades estiguin **omplertes**, que siguin les **correctes** a les quals estan registrades a la base de dades, i que compleixin els **requisits de seguretat**, com ara els de la contrasenya.

Similar passa amb els formularis **d’enviament de formulari** per canviar la contrasenya d’usuari en cas que s’hagi oblidat i el formulari per **canviar la contrasenya**. Aquestes sempre fan les **comprovacions necessàries** per que les dades compleixin els requisits ja nombrats.

***Diagrama* *complet*:**



### Usuari registrat:

El funcionament principal a l’hora de carregar la **pàgina principal** es comprovar els projectes emmagatzemats per **l’usuari** que a iniciat sessió, un cop s’obté totes les dades de cada projecte, es mostra com a **elements HTML a seleccionar** amb les **opcions disponibles** de cada projecte a la pàgina d’inici.

Un cop es carrega la pàgina d'inici, l'usuari pot començar a **crear**, **editar** o **canviar les dades** d'un projecte (**eliminar**, **canviar el nom** del projecte), **duplicar** un projecte, **visualitzar** **la informació** d'un projecte, obrir la **configuració** d'usuari, o **tancar la sessió**.

Quan es vol **crear un projecte**, aquest mostra les opcions de començar de zero a partir d'una **base fixa**, o **importar** un projecte d'un arxiu local **JSON**. En els dos casos es processa les dades del contingut i es guarden a la base de dades creant un nou camp amb un **identificador únic**, un cop realitzat aquesta acció, es passa al controlador per **desserialitzar** les dades com a objectes **JavaScript** i mostrar el codi **HTML** per començar a editar el projecte.

Aquesta última acció passa també amb la **selecció d'un projecte ja creat**, ja que és necessari primer obtenir totes les dades del projecte a partir de **l’identificador**, processar-les i mostrar-les.

Com a opcions **d'edició d'un projecte**, si a un projecte es **guarda els canvis**, passa a realitzar la **desserialització** per guardar les dades a la **base de dades** i tornar a mostrar el contingut a la pàgina.

En cas que es vulgui **visualitzar el projecte**, a l'hora de clicar el botó de **visualitzar**, també es realitza l'acció de **guardar qualsevol canvi** realitzat al projecte i guardar-ho a la **base de dades**, i torna a obtenir les dades per processar-les de forma que **amaga les opcions d'edició**.

Quan es vol **descarregar el projecte**, aquest realitza la **desserialització** i dona l’opció d’escollir on guardar l’arxiu JSON.

El procés a l'hora de voler **eliminar**, **canviar el nom** o **duplicar** un projecte són similars, ja que en cada cas s'obté **l'identificador** del projecte seleccionat i es realitza l'acció desitjada a partir d'una sentència **SQL** a la **base de dades**.

Quan un usuari vol **editar el nom** del seu compte, es comprova que el camp en el formulari **no estigui buit** i que el nou nom seleccionat no estigui **ja registrat a la base de dades**. Un cop tot sigui correcte, es realitza el canvi a la **base de dades** i torna a mostrar la pàgina amb el **nou nom** i el **missatge de confirmació**.

Similar passa amb el formulari de **canvi de contrasenya**, primer es comprova que **cap camp estigui buit**, i després si es compleix els **requisits de seguretat** per la contrasenya. En cas de que tot sigui correcte, es realitza el canvi a la **base de dades** i torna a mostrar la pàgina amb el **missatge de confirmació**.

***Diagrama* *complet*:**

Interfaz de usuario gráfica

Descripción generada automáticamente

# Interfície d’usuari

Disseny d’interfície d’usuari realitzat amb **Canva.**

### Wireframe usuari anònim:

* Pàgina d’inici anònim:

**Interfaz de usuario gráfica, Aplicación

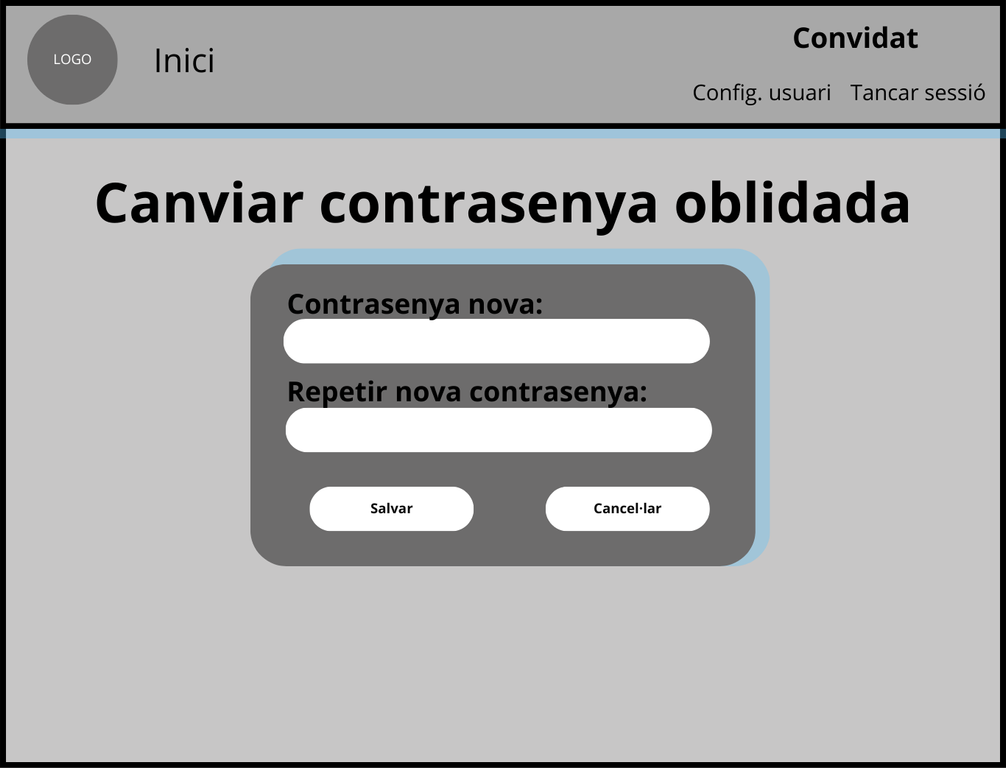
Descripción generada automáticamente**

* Formulari iniciar sessió:

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

* Formulari canviar contrasenya oblidada:

****

* Formulari registrar-se:

****

### Wireframe usuari registrat:

* Pàgina Inicial usuari registrat:

Interfaz de usuario gráfica

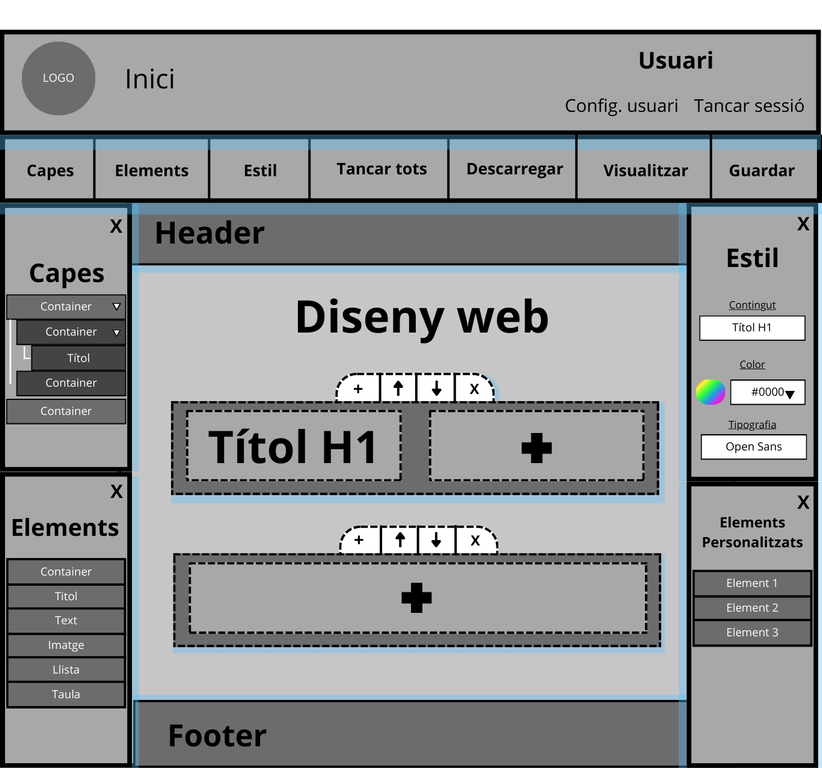
Descripción generada automáticamente

* Pàgina de selecció de base i importar Projecte:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Edició d’un projecte com a usuari:



* Pàgina de configuració del compte:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Formulari per canviar la contrasenya de l’usuari:

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

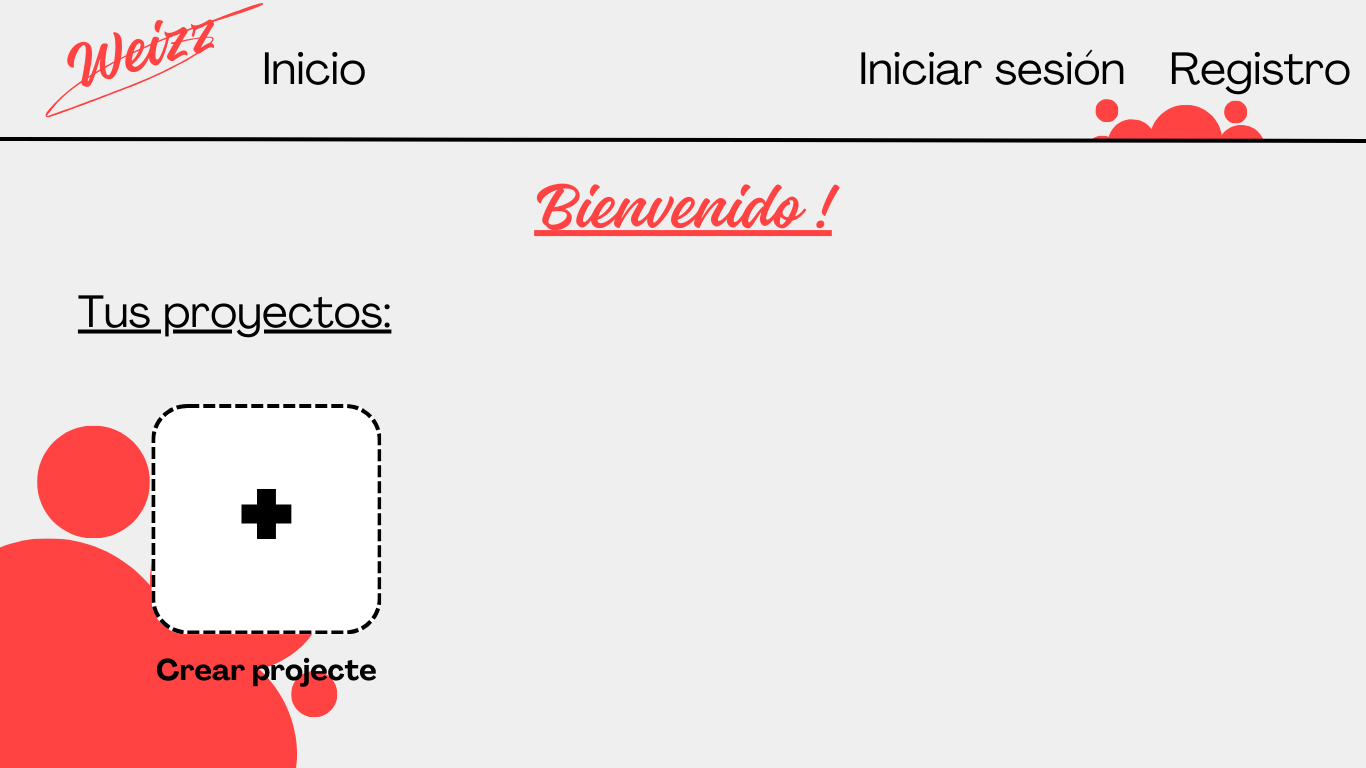
Descripción generada automáticamente

* Formulari per canviar el nom de compte:

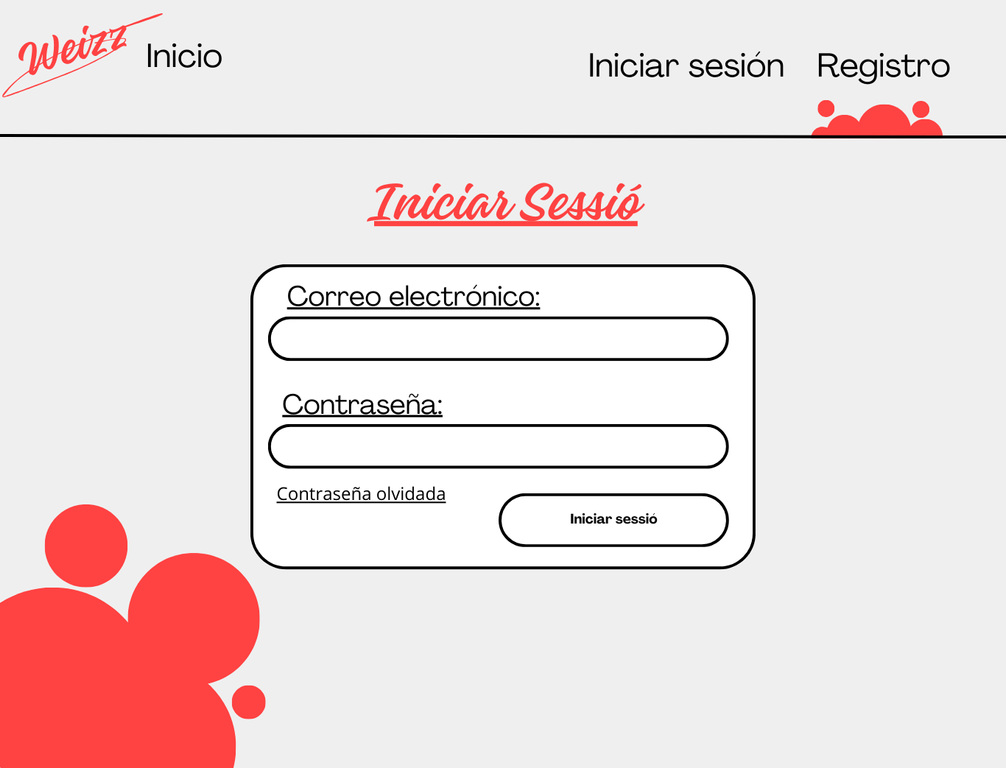


### Mockup usuari anònim:

* Pàgina d’inici anònim:



* Formulari iniciar sessió:



* Formulari registrar-se:

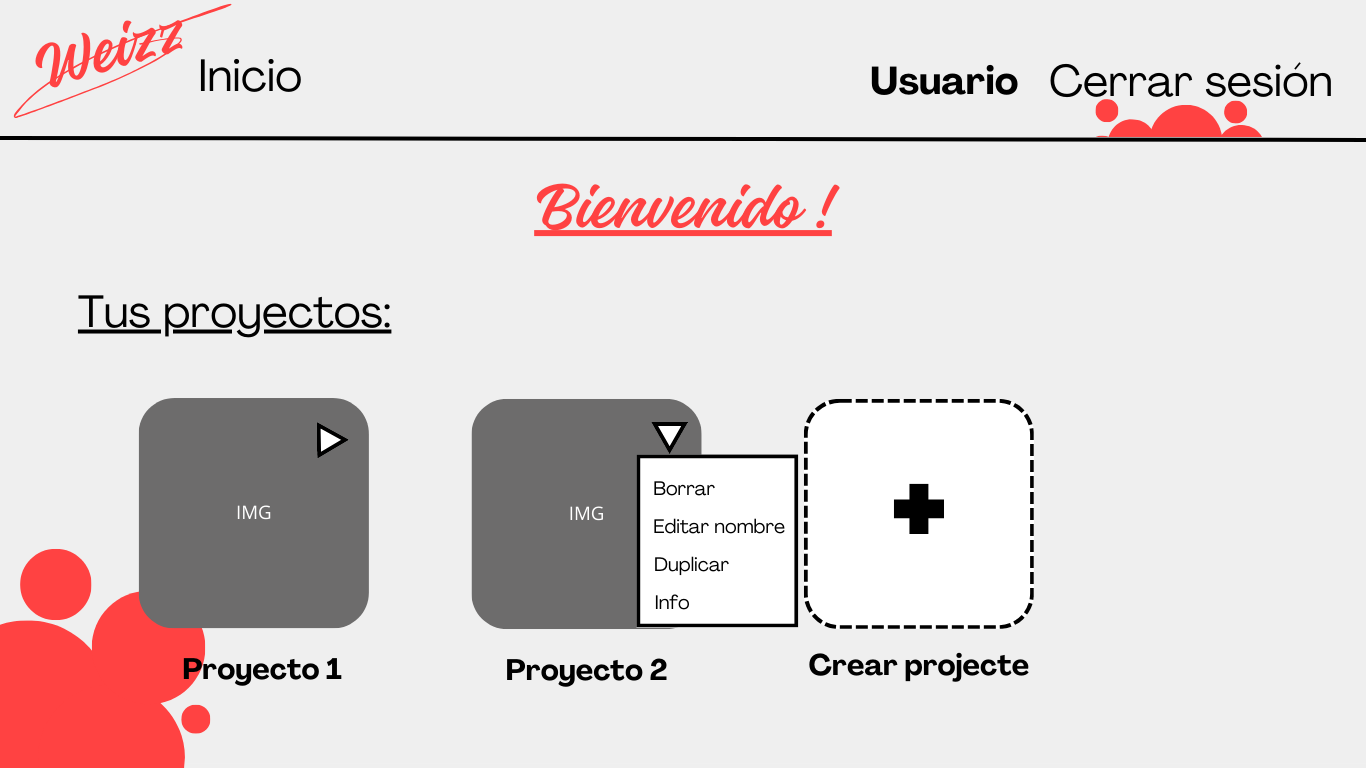


* Formulari canviar contrasenya oblidada:

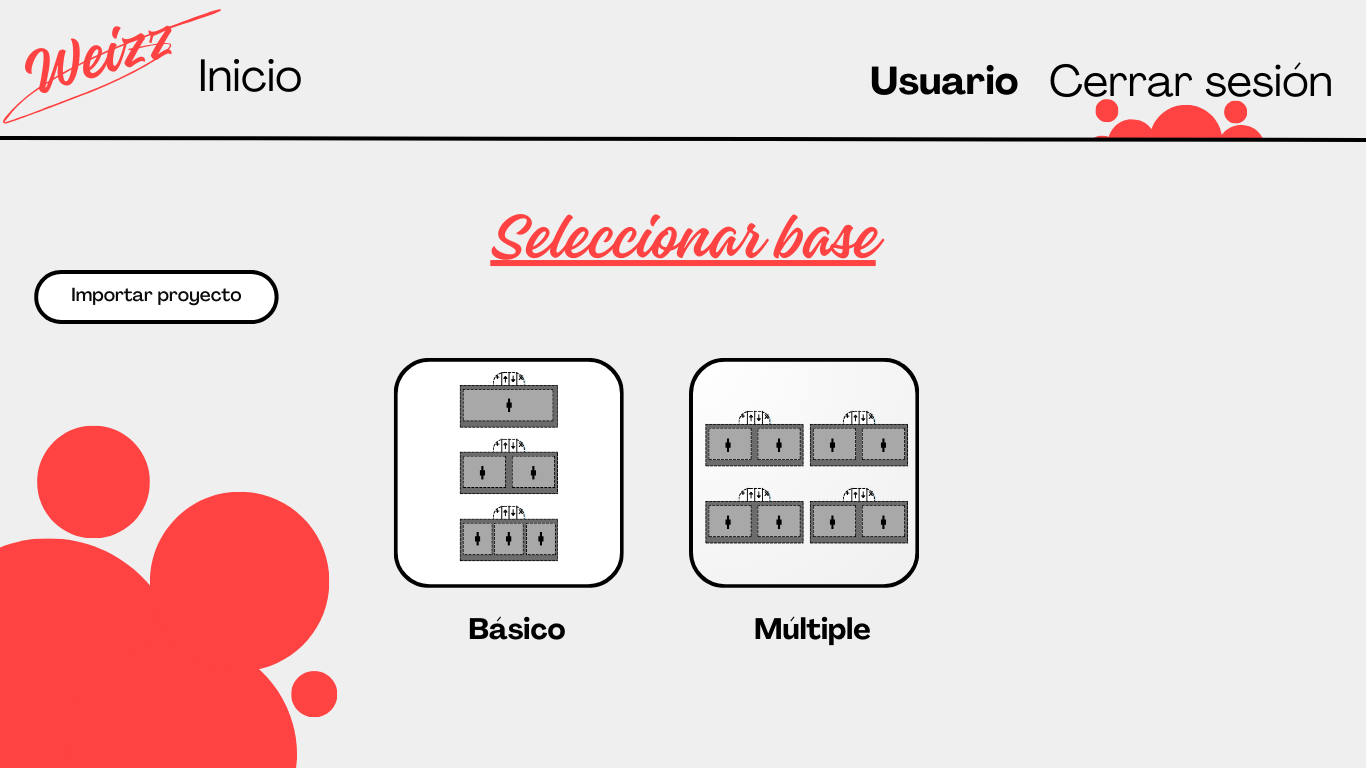


### Mockup usuari registrat:

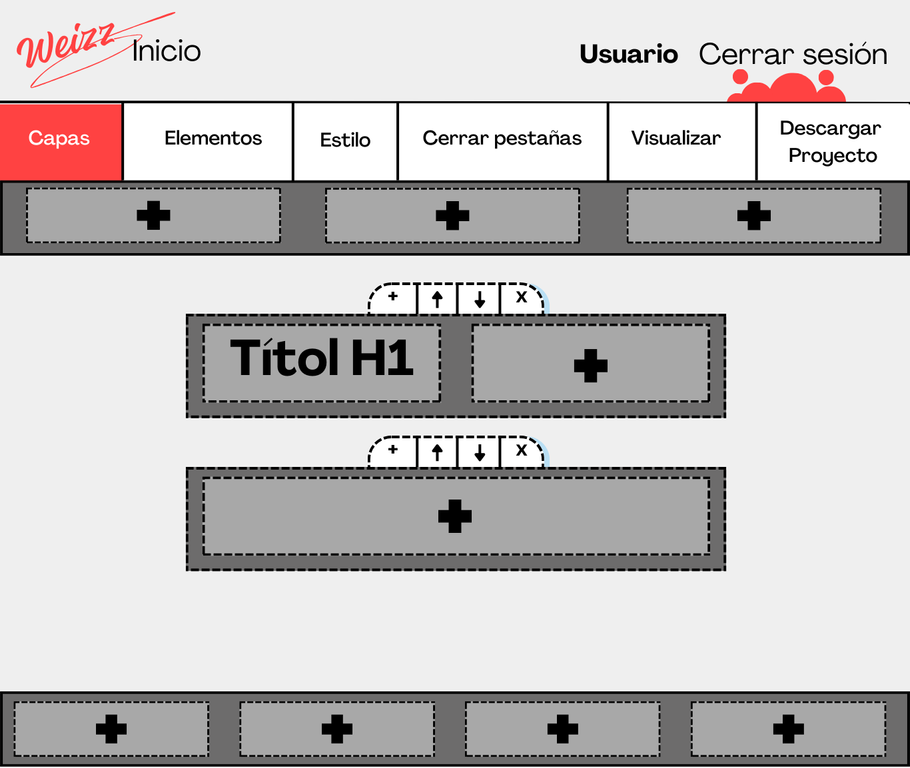
* Pàgina Inicial usuari registrat:



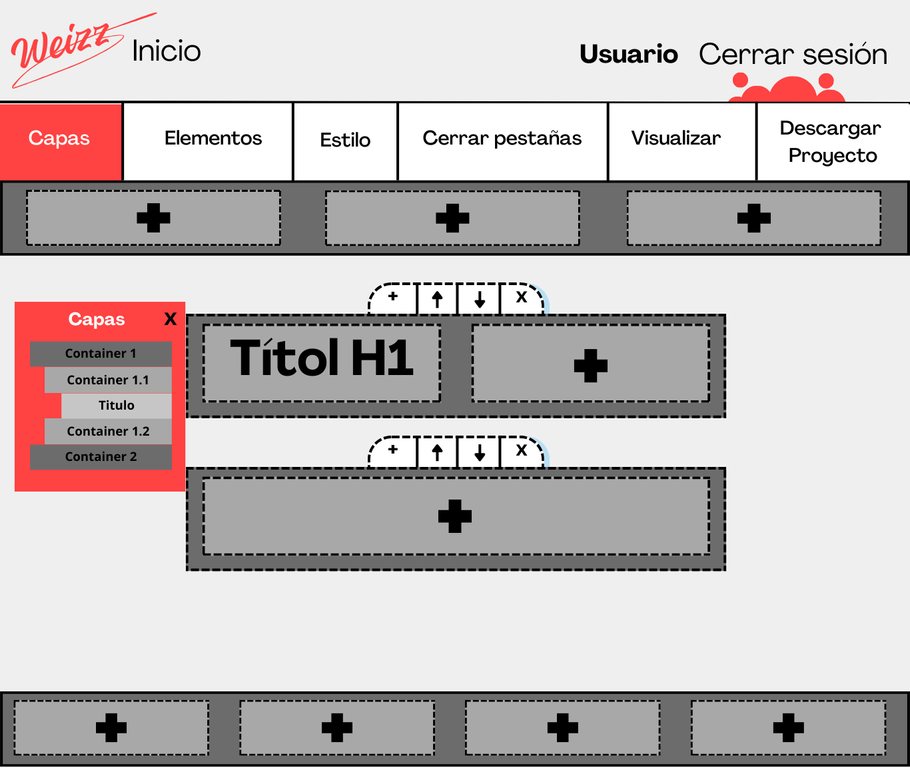
* Pàgina de selecció de base i importar Projecte:



* Edició d’un projecte com a usuari:



* Edició d’un projecte com a usuari amb requadre de capes oberta:



* Pàgina de configuració del compte:



* Formulari per canviar la contrasenya de l’usuari:



* Formulari per canviar el nom de compte:



# Diagrama Base de dades:

La base de dades es creat amb **PHPMyAdmin**.

A primera vista i com a primera versió del projecte he vist convenient crear dues taules en la base de dades de la pàgina.

### Usuaris:

La taula principal es la taula d'usuaris, on son guardats les seves dades i altres que no es mostraran a vista de l’usuari, però sí que seran necessàries pel seu funcionament Back End. Aquestes últimes són l'identificador principal de l'usuari en la base de dades i el token, que servirà per crear seguretat a l'hora de voler canviar la contrasenya, que **canviarà** cada vegada que l'usuari ho sol·liciti.

Tabla

Descripción generada automáticamente

### Projectes:

En la taula de projectes, per ara, es necessari un identificador propi, l'identificador de l'usuari que ha creat el projecte (Foreign key), i el contingut JSON complet del projecte. Aquest inclou totes les **característiques** de cada element aplicat al projecte.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

***Diagrama complet***:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

# Implementació:

## Front End:

1. La programació Front End en la **pàgina inicial** funciona mostrant els projectes creats al compte d'usuari. És a dir, si un usuari ja registrat tenia projectes creats, el programa Front End s'encarregarà de demanar el projectes d’aquest usuari a la base de dades y mostrar els **elements de selecció** de cada projecte. A més, es mostra el nom d'usuari i les opcions de la pàgina, com anar a la **configuració del compte**, **tancar sessió**, **inici**, i **pàgina de projectes**.

La funcionalitat principal del Front End es troba a l'hora d'estar **editant un projecte web**, ja que aquest gestiona la creació d’objectes JavaScript i elements HTML perquè concordi amb la vista que vol l'usuari. Aquest sempre treballarà amb classes JS i elements **DOM (Model d'objectes del document)** amb funcionalitats de **Bootstrap** per un disseny atractiu per l'usuari. És a dir, sempre es crean elements HTML amb les propietats adequades de Bootstrap.

1. El mateix passa amb **l'estil** de cada element, ja que facilita la configuració dels estils amb un requadre flotant que dinàmicament modifica el **codi CSS** i les **propietats JS** de cada objecte.

**L’estructura i construcció** de cada projecte i els elements que s’implementen a dins d’aquest es basa en les propietats i funcions de cada classe JS creat per cadascun d’aquest. Cada classe conté unes propietats específiques per modificar l’estil i tornar a crear l’element DOM que es mostrarà al projecte.

Per exemple, una de les propietats de l'objecte **Llista** seria la llista de variables String que conté, el color de la lletra, el nombre d’apartats que conté, entre altres.

Aquestes propietats seran especificades **al diagrama de classes** del projecte.

1. La funcionalitat per poder visualitzar totes les capes del projecte que s'està editant és fer un bucle amb un **control d'esdeveniments** de cada element a dins del cos, header o footer del projecte. Per a cada element trobat en el control d'esdeveniment, s'aplicarà a la **visualització de menú de capes**. Aquest s'actualitzarà per cada element **esborrat o afegit** al projecte.

També es pot **seleccionar** un element desitjat a partir d'aquest menú per poder editar-ho.

1. Una altra implementació a cada editor de projecte són els esdeveniments aplicats per facilitar l’aplicació d’elements a cada Container. Són múltiples els tipus d’esdeveniments aplicats a les diferents opcions garantits del projecte:
   1. Cada element o container té aplicat un event listener a l’hora de **clicar** a sobre d’aquests, detectant si el **container està buit**, o si hi té un **element dins d’aquest**. En cas que estigui buit, s’obrirà la pestanya dels **elements disponibles**. En cas que hi tingui un element dins, s’obrirà la pestanya de les **opcions d’estil** de l’element seleccionat.

A més, en cada zona del projecte (header, body, footer), conté un botó per obrir la **selecció de color de fons** que es vulgui aplicar.

* 1. Cada bloc de containers conté 4 opcions per modificar les ubicacions d’aquests.

En aquestes opcions existeix la possibilitat de **crear** un altre bloc amb el nombre seleccionat de containers dins d’aquest.

**Moure** el bloc de containers de posició. Depenent de la posició d’aquest es mourà a l’**esquerra i dreta**, o **amunt i avall**...

I **eliminar** permanentment el bloc de containers seleccionat amb els elements que conté.

* 1. Cada container conté event listeners a l’hora de detectar quan un element **passa a sobre** d’aquest (ja que **l’estil** del container canvia) i quan es “**deixa caure**” en el container seleccionat (l’element **s’aplica** dins del container).

Aquesta opció “**drag & drop**” facilita la creació d’elements.

Similar passa quan l’element a arrossegar és un element **Container**, les opcions canvien i l’estil s’aplica als costats del bloc de container. Això, permet generar més blocs de containers en un mateix fila.

1. També s’ha implementat amb JavaScript l'opció de **descarregar** un projecte com arxiu JSON, on **s'emmagatzema tot el contingut editat**.
2. L'opció de **visualitzar** el projecte sense els elements d'edició funciona de manera que JavaScript agafa tots els elements implementats per l'usuari i els aplica a un nou arxiu ja creat prèviament perquè mostri la pàgina **amagant els objectes de containers o d'edició**.
3. Sempre que un usuari vulgui crear un compte, es fa la comprovació necessària per complir un mínim de característiques restrictiu com ara la **contrasenya**, ja que ha de tenir un **mínim de 8 caràcters amb una majúscula i un caràcter numèric**.

## Back End:

1. El Back End s'encarrega principalment de processar les dades sol·licitades per **mostrar els projectes emmagatzemats** pels usuaris, a més de guardar el registrament d'un nou usuari i projecte.
2. El funcionament principal de com guardar un projecte a la base de dades, és emmagatzemar tot el contingut HTML com a **JSON** en un dels camps de la taula projectes.

És un camp a la base de dades que **únicament** s'ompliria de **text**, i que a l'hora de ser demanat pel Front End, fa el procés necessari per tornar a mostrar el projecte adequadament.

1. Cada funcionalitat de modificar un compte d'usuari, com ara la contrasenya o el nom, i obtenir totes les seves dades, seran a partir dels **funcionaments i peticions MySQL** amb la llibreria **PDO**. Aquestes funcionalitats faciliten el correcte funcionament d’intercanvi de dades i d'obtenció d'aquestes a l'hora de mostrar totes les dades d'un únic usuari.

# Planificació temporal

1. Creació Mockup. (Canva) ***1 dia***
2. Diagrama activitat d'ús. (draw.io) ***1 dia***
3. Creació diagrama de classes. (draw.io***) 1 o 2 dies***
4. Creació de base de dades, i de taules necessàries. (PHPMyAdmin/MySQL)

Introducció d’elements bàsics a la base de dades (taula elements\_basics). (PHPMyAdmin/MySQL) ***1 dia***

1. Configuració principal d'usuari: ***7 dies en total***
   1. Desenvolupament d'estructura principal de la pàgina principal. (HTML)
   2. Desenvolupament d'estructura principal de la pàgina d'inici de sessió. (HTML)
   3. Desenvolupament d'estructura principal de la pàgina de registrar usuari. (HTML)
   4. Desenvolupament d'estructura principal de la pàgina de canviar nom d'usuari. (HTML)
   5. Desenvolupament d'estructura principal de la pàgina de canviar contrasenya. (HTML)
   6. Desenvolupament d'estructura principal de la pàgina d'opcions de configuració d'usuari. (HTML)

***Les lletres anteriors (de la a. a la f.) les desenvoluparé conjuntament en un total de 2 o 3 dies.***

* 1. Funcionament principal de registrar usuari. (PHP/MySQL)
  2. Funcionament principal d'iniciar sessió. (PHP/MySQL)

***Les lletres anteriors (de la g. a la h.) les desenvoluparé conjuntament en un total de 1 dia.***

* 1. Funcionament principal de canviar nom d'usuari. (PHP/MySQL)
  2. Funcionament principal de canviar contrasenya d'usuari. (PHP/MySQL)

***Les lletres anteriors (de la i. a la j.) les desenvoluparé conjuntament en un total de 1 dia.***

* 1. Comprovació de correcte funcionament.
  2. Fixar falles trobades.

***Les lletres anteriors (de la k. a la l.) les desenvoluparé conjuntament en un total de 1 o 2 dies****.*

1. Creació de plantilles base de projectes (Quatre en total). (HTML) ***1 o 2 dies***
2. Configuració principal projectes: ***37 dies en total***
   1. Aplicació de plantilles base al projecte. (JavaScript) ***2 dies***
   2. Creació de classes d'elements base (Container, Títol, Text, Imatge, Llista, etc.). (JavaScript) ***3 dies***
   3. Configuració d’aplicació de container principal, i configuració d'aquesta. (JavaScript) ***4 dies***
   4. Configuració d’aplicació d'elements als containers i configuració "drag and drop". (JavaScript) ***4 o 5 dies***
   5. Configuració d'edició de “header” a la pàgina (canvia depenent de la base). (JavaScript) ***3 dies***
   6. Configuració d'edició de “footer” a la pàgina (canvia depenent de la base). (JavaScript) ***2 dies***
   7. Configuració de canvi d'estil per a cada element. (JavaScript) ***4 o 5 dies***
   8. Configuració de desar el projecte a la base de dades. (JavaScript/PHP/MySQL) ***2 dies***
   9. Configuració de vista de les capes del projecte. (JavaScript) ***3 dies***
   10. Configuració per visualitzar el projecte en una altra pàgina. (JavaScript) ***2 dies***
   11. Configuració per descarregar projecte (JavaScript) ***1 o 2 dies***
   12. Comprovació de correcte funcionament.
   13. Fixar falles trobades.

***Les lletres anteriors (de la l. a la m.) les desenvoluparé conjuntament en un total de 3 o 4 dies****.*

1. Importació de projectes externs (JavaScript/PHP/MySQL) ***2 dies***
2. Configuració principal elements: ***14 dies en total***
   1. Desenvolupament d'estructura principal de la pàgina de creació/edició. (HTML)
   2. Desenvolupament d'estructura principal de la pàgina de visualització d'elements personalitzats. (HTML)

***Les lletres anteriors (de la a. a la b.) les desenvoluparé conjuntament en un total de 1 dia****.*

* 1. Configurar canvi d'opcions d'estil depenent del tipus d'element a l'hora d'editar l'element personalitzat. (JavaScript) ***2 o 3 dies***
  2. Configurar l’emmagatzematge de l’element personalitzat a la base de dades. (PHP/MySQL) ***2 dies***
  3. Configurar la visualització d'elements personalitzats a la pàgina d’elements personalitzats. (PHP/MySQL/JS) ***1 o 2 dies***
  4. Visualització d’elements personalitzats a un projecte i correcte funcionament. (PHP/MySQL/JavaScript) ***3 o 4 dies***
  5. Comprovació de correcte funcionament.
  6. Fixar falles trobades.

***Les lletres anteriors (de la g. a la h.) les desenvoluparé conjuntament en un total de 3 o 4 dia****.*