

# LLIBRE INTERACTIU

Creant un llibre interactiu amb Ionic

## DESCRIPCIÓ BREU

Com crear pas a pas una aplicació per a llibres interactius i com utilitzar les plantilles per crear-ne nous llibres interactius per la aplicació

## GACS

Ionic

## Contenido

Aplicació Book .....	2
Iniciant l'aplicació.....	2
Modificant els noms.....	3
Modificant una pàgina .....	4
Crear una pàgina .....	4
L'aplicació no va .....	6
Creant un llibre.....	8
Introducció .....	8
Creant un llibre.....	8
Inserint la portada .....	8
Accedint al llibre.....	8
Emprant enllaços.....	9
Emprant botons.....	9
Redimensionant botó i imatge .....	10
Editant el llibre .....	11
Accedint al llibre des de la portada.....	12
Accedint al llibre mitjançant un botó.....	13
Creant una pàgina .....	13
Creant un llibre amb les plantilles.....	21
Com crear un llibre nou.....	21
Afegint el nou llibre a l'aplicació .....	21
Creant la portada .....	22
Creant una pàgina .....	23
Com afegir un text.....	23
Com afegir una imatge .....	23
Com afegir una animació.....	24
Com afegir un vídeo .....	24
Com afegir un so .....	25

## Aplicació Book

### Iniciant l'aplicació

Aquesta aplicació per a dispositius mòbils s'ha creat amb la plantilla **sidemenu** del **framework Ionic** i utilitzar el **framework Angular** per a la part visual. Per crear la plantilla s'ha obert una línia de comandament o terminal i s'ha teclejat el següent comandament:

---

```
$ionic start Book sidemenu --type=angular
```

---

Un cop creat el projecte, es recomana la seva execució al navegador per veure l'aspecte inicial. Per fer això primer entrarem a la carpeta del projecte amb el comandament

---

```
$cd Book
```

---

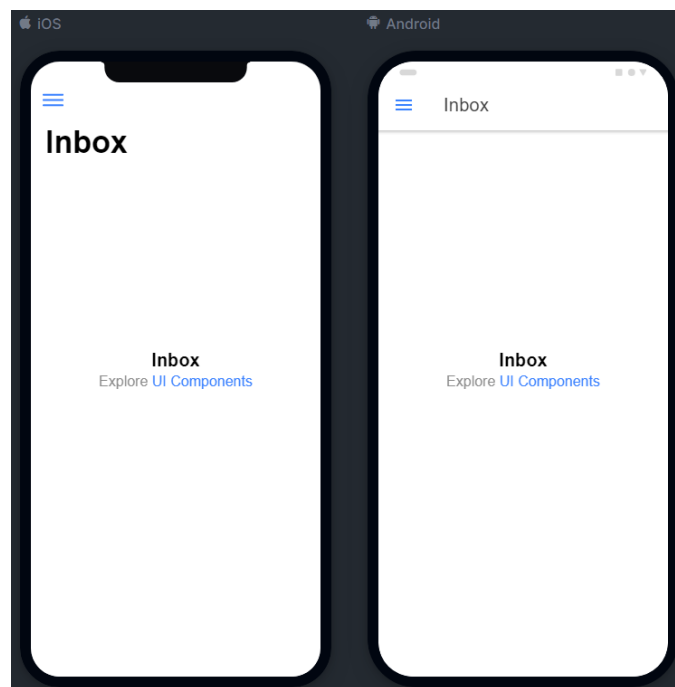
I un cop dins de la carpeta del projecte escriurem el comandament per a executar l'aplicació:

---

```
$ionic serve -l
```

---

Si ho hem fet de forma correcta se'ns mostrarà al navegador la Figura 1.



*Figura 1. Aspecte inicial de l'aplicació.*

Com es pot veure a la Figura 1 a la cantonada superior esquerra hi ha un menú desplegable que al pitja sobre ell mostra la Figura 2.

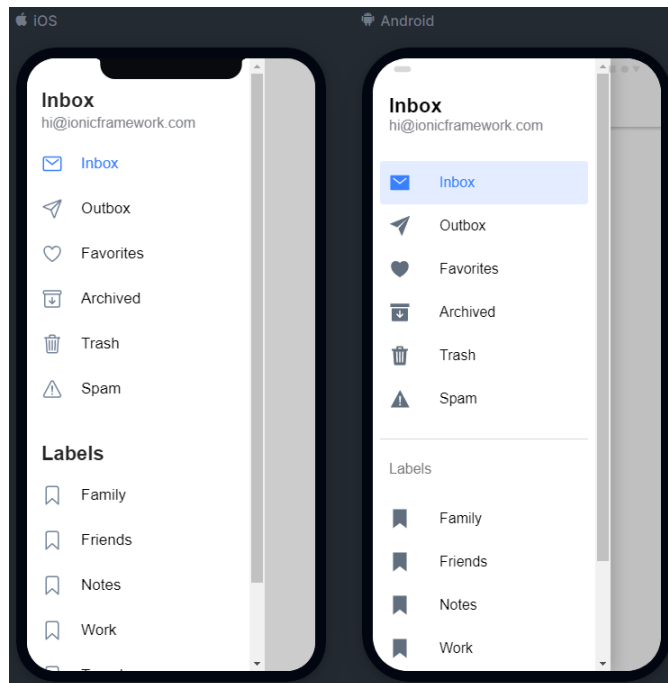


Figura 2. Menú lateral desplegat.

Com es pot observar a la Figura 2 ja hi ha definits per defecte una sèrie de menús i enllaços a diferents pàgines, el següent pas serà modificar el nom i esborrar els enllaços que no ens facin falta.

### Modificant els noms

Per modificar els noms i les icones que apareixen al desplegar el menú lateral (**sidemenu**), hem d'obrir l'arxiu **src/app/app.component.ts**. D'aquest arxiu, només hem de prestar atenció al text que està dins de **appPages**, que és un **array**.

```
public appPages = [
  { title: 'Inbox', url: '/folder/Inbox', icon: 'mail' },
  { title: 'Outbox', url: '/folder/Outbox', icon: 'paper-plane' },
  { title: 'Favorites', url: '/folder/Favorites', icon: 'heart' },
  { title: 'Archived', url: '/folder/Archived', icon: 'archive' },
  { title: 'Trash', url: '/folder/Trash', icon: 'trash' },
  { title: 'Spam', url: '/folder/Spam', icon: 'warning' },
];
```

Aquí el que observem és el següent:

- **title.** És el títol que apareix a l'enllaç del menú.
- **url.** És la direcció de la pàgina.
- **icon.** És la icona que hi ha associada amb la pàgina al menú.

Serà en aquest arxiu on introduïrem els títols dels diferents llibres, como veurem més tard. Ara el que farem serà modificar els tres primers títols per Inic, Guardar, Carregar i El-meu-llibre. L'**array** quedarà de la següent manera:

```
public appPages = [
  { title: 'Inici', url: '/pages/home', icon: 'home' },
  { title: 'Guardar', url: ' /save', icon: 'bookmark' },
  { title: 'Carregar', url: ' /load', icon: 'archive' },
  { title: 'El meu llibre', url: '/el-meu-llibre', icon: 'book' },
  { title: 'Trash', url: '/folder/Trash', icon: 'trash' },
  { title: 'Spam', url: '/folder/Spam', icon: 'warning' },
];
```

Pel moment, deixarem dos títols que podrem modificar per a nous llibres, així com les etiquetes (**labels**), que poder eliminarem més tard.

Ja tenim el menú lateral modificat al nostre grup, ara hem de començar a crear les pàgines de l'aplicació on podem realitzar totes les accions.

### Modificant una pàgina

El següent pas serà modificar una pàgina, la pàgina **folder** que és la que es mostrarà al obrir l'aplicació. Podríem crear una pàgina **home** on fer totes aquestes modificacions; però aprofitarem l'existència de l'arxiu **folder** i així no tenir que fer més modificacions de les necessàries.

Les modificacions visual les realitzarem a l'arxiu **src/app/folder/folder.page.html** on modificarem el text que hi ha a l'etiqueta **<ion-title>** per **Inici** i esborrarem tot el que hi ha dins de **<ion-content>**. En aquesta pàgina mostrarem els llibre que hem creat per a la nostra aplicació, així que veurem més tard com crear un nou llibre a la secció TODO!!

### Crear una pàgina

Anem a crear una pàgina per a emmagatzemar posició de l'usuari dins del llibre. Així que anirem a un terminal o línia de comandaments i escriurem el següent comandament dins de la carpeta del nostre projecte:

---

```
$ionic g page pages/save
```

---

Per veure si hem creat de forma correcta els enllaços del menú lateral i la pàgina obrirem el projecte a un navegador i anirem a l'opció carregar. Si tot ha anat bé deuríem observar la Figura 3.

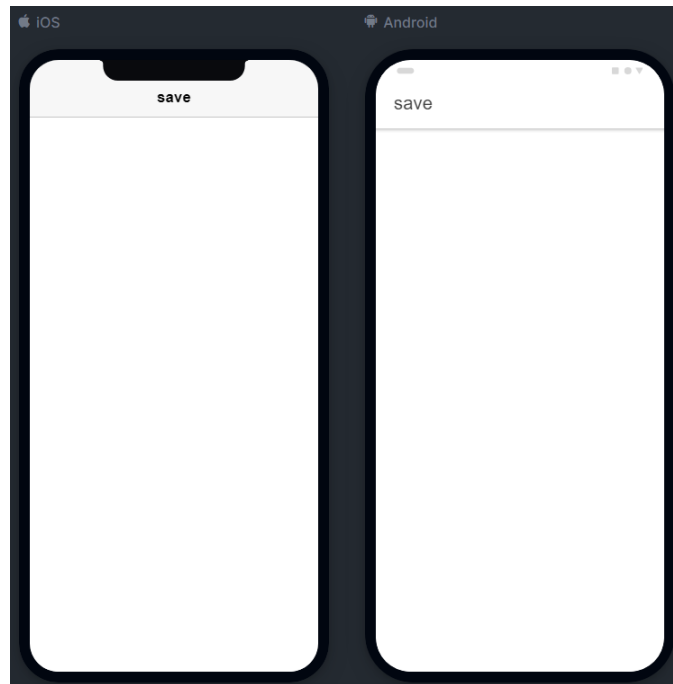


Figura 3. Pàgina per guardar la posició.

Actualment la vista que tenim està en blanc i només mostra el títol, així que comencem a donar contingut a aquesta pàgina. Per fer-ho obrirem l'arxiu **src/app/pages/save/cave.page.html** i començarem per modificar el títol de la pàgina per **Guardar** i afegint el menú lateral que no apareix a aquesta pàgina. L'arxiu hauria de quedar de la següent manera:

```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-menu-button></ion-menu-button>
    </ion-buttons>
    <ion-title>
      Guardar
    </ion-title>
  </ion-toolbar>
</ion-header>

<ion-content>

</ion-content>
```

L'altre modificació que realitzarem serà la de crear un botó per a guardar la lectura del llibre. Per fer-ho inclourem el següent text a l'etiqueta **<ion-content>**.

```
<ion-content>
  <ion-button expand="block" (click)="save()">Guardar</ion-button>
</ion-content>
```

Aquest botó ara mateix només crida a un mètode anomenat **save()** que és el que utilitzarem per emmagatzemar la nostra lectura. Per que el botó no quedi tan enganxat a la part superior hem de retocar els estils.

L'arxiu d'estils d'aquesta pàgina és **src/app/pages/save/save.page.css**, l'obrim i afegim les següent línies.

```
ion-button{
  margin-top: 5vh;
}
```

Amb aquesta línia s'està indicant que volem que el marge superior del botó sigui del 5% de l'alçada de la pantalla.

Ara hem de modificar l'arxiu **src/app/pages/save/save.page.ts** per implementar el mètode al que crida el botó que hem creat a la vista. Obrim l'arxiu per realitzar les modificacions oportunes.

Per emmagatzemar-la utilitzarem el **localStorage** d'Ionic. Per utilitzar-ho creem el mètode **save()** i el definim de la següent manera.

```
save(){
  localStorage.setItem('key', this.page);
}
```

Aquest mètode utilitza una variable anomenada **page**; per tant, hem de definir la variable per poder utilitzar-la. Per aquest motiu, abans del mètode **constructor()** escrivim la següent línia;

```
page:string="0";
```

Amb aquesta línia estem declarant **page** com a una variable de tipus **string** i estem inicialitzant-la al mateix temps amb el valor **0**.

### L'aplicació no va

Si anem a un navegador i intentem executar l'aplicació veurem que no podem tornar a la pàgina inicial des de la pàgina **Guardar**. Per tant, l'aplicació no funciona de forma correcta. Per solucionar aquest problema hem d'anar a l'arxiu **src/app/app-routing.module.ts** i fer algunes modificacions als **paths**. Després de les modificacions l'**array routes** ha de tenir el següent aspecte:

```
const routes: Routes = [
  {
    path: '',
    redirectTo: 'folder',
    pathMatch: 'full'
  },
  {
    path: 'folder',
    loadChildren: () => import('./folder/folder.module').then( m =>
m.FolderPageModule)
  },
]
```

```
{
  path: 'save',
  loadChildren: () => import('./pages/save/save.module').then( m =>
m.SavePageModule)
}
];
```

Ara si tornem a un navegador veurem que l'aplicació funciona de forma correcta.



## Creant un llibre

### Introducció

En aquest capítol anem a veure com crear un nou llibre des de zero, Al capítol, també es veurà com crear la lògica de les pàgines i crearem diferents plantilles que es podran utilitzar per un llibre. Les plantilles que es veuran son:

- Només text.
- Text amb una imatge.
- Text amb una animació.
- Text amb un vídeo.
- Text amb un so.

### Creant un llibre

El primer que farem per crear un llibre serà obrir un terminal o una línia de comandament a la carpeta del nostre projecte i escriure el següent comandament:

---

```
$ionic g page llibres/el-meu-llibre
```

---

Amb això creem una nova pàgina que durà el nom del llibre que volem crear. A aquesta carpeta serà on creem totes les pàgines del nostre llibre.

El proper pas és crear una portada pel llibre. Per això hi ha una plantilla de PowerPoint que permet la creació de portades. La plantilla es troba a la carpeta web. Si es vol utilitzar un altre forma de creació de portada com ara [Canva](#)<sup>1</sup> es pot fer. La portada que creem la guardarem en un format d'imatge com ara: jpg, jpeg, png, tiff, gif,... a la carpeta **img/el-meu-llibre** que haurem creat a la següent ruta: **src/app/assets**.

Es recomana que la imatge tingui el mateix nom que el llibre i que creem una carpeta amb el nom del nostre llibre si anem a utilitzar imatges al llibre. D'aquesta forma tindrem tot més endreçat.

### Inserint la portada

Per inserir la portada que ens donarà accés al llibre obrirem el fitxer

**src/app/folder/folder.page.html** i escriurem el següent dins de l'etiqueta **<ion-content>**:

```

```

Si guardem i mirem el resultat a un navegador veurem que hi ha diversos problemes:

- La imatge ocupa gairebé tota la pantalla.
- El lector detecta la imatge però no diu res.
- Al pitjar sobre la imatge no s'obre el llibre.

### Accedint al llibre

Podem accedir al llibre utilitzant diverses estratègies com ara:

- Enllaç.

---

<sup>1</sup> La web i la forma de creació és inaccessible pels lectors de pantalla.

- Botó.

### Emprant enllaços

La tècnica d'utilitzar un enllaç per obrir un llibre i vincular una pàgina amb una altre és possiblement la més senzilla al no necessitar res de codi de programació i utilitzar les etiquetes d'**HTML**.

Si volem utilitzar aquesta fórmula, hem d'afegir una etiqueta **<a>** abans de la **<img>**. El codi seria el següent:

```
<a routerLink="/el-meu-llibre" title="Obrir el meu llibre">
  
</a>
```

Si hem realitzar tot de forma correcte se'ns mostrà una pàgina en blanc amb el títol **el-meu-llibre**.

### Emprant botons

Si ens decidim per utilitzar botons, llavors hem de teclejar una mica de codi, però les opcions que ens donen és molt superior a la de un senzill enllaç.

El codi del fitxer **src/app/llobres/el-meu-llibre/el-meu-llibre.page.html** ha d'incloure un botó on es cridarà un mètode. El codi que hem descriure és el següent:

```
<button (click)="openBook(0)">
  
</button>
```

Pel que fa al mètode **openBook( )**, l'hem de declarar i definir dins el fitxer **src/app/llobres/el-meu-llibre/el-meu-llibre.ts**. En aquest fitxer declararem i inicialitzarem un **array** amb el nom de cadascun del llibres que creem entre cometes i separats per una coma de la següent forma:

```
llibres:string=["/el-meu-llibre"];
```

Aquest **array** contindrà tots els títols dels diferents llibres i serà cridat pel mètode **openBook( )**. Per crear aquest mètode primer importarem i injectarem el mòdul **Router**. La importació la farem afegint una línia sota la primera línia del fitxer de la següent forma:

```
import { ActivatedRoute, Router } from '@angular/router';
```

Un cop importat el mòdul l'hem d'injectar al constructor de la següent manera:

```
constructor(private router: Router) { }
```

Ara ja podem crear el mètode sense cap tipus de problema de la següent manera:

```
openBook(a:number){
  this.router.navigate([this.llibres[a]]);
}
```

A aquest mètode li passem un **atribut** des de la vista amb **openBook(0)**. El valor 0 marca la posició de l'array que ocupa el llibre. Això ens permet que al crear un nou llibre no tinguem

que crear un nou mètode, sinó que podem cridar al mateix amb una altre **paràmetre**, per exemple: **openBook(1)** per que el botó obri el llibre que ocupa la segona posició de l'**array**.

Si utilitzem aquest mètode, veiem que no es mostra de forma correcta la portada del llibre. Veiem com arreglar-ho al següent apartat.

### Redimensionant botó i imatge

Per redimensionar la imatge de la portada i el botó hem de retocar el fitxer **src/app/llobres/folder/folder.page.scss**.

Però abans modificarem l'arxiu **src/app/folder/folder.page.html** per fer la imatge més accessible i que els lectors de pantalla no trobin la imatge i només trobin un botó amb l'enllaç pertinent per obrir el llibre.

```
<button (click)="openBook(0)" data-message="Obrir el-meu-llibre"
type="button" class="btn">
  <div id="el-meu-llibre" aria-hidden="true" viewBox="0 0 10 10"></div>
</button>
```

Com podem observar ara no hi ha una etiqueta **<img>** sinó una **<div>** que és la que contindrà la imatge com a fons (**background**). D'aquest forma s'evita que els lectors de pantalla identifiquin les imatges quan aquestes son purament decoratives. Per fer això hem de modificar el fitxer d'estils **src/app/folder/folder.page.scss**.

Per això obrirem l'arxiu esborrarem tot el contingut a excepció de **ion-menu-button** i el que conté entre claus i escriurem el següent:

```
#el-meu-llibre{
  margin: 2vh 5vw;
  height: 40vh;
  width: 80vw;

  background: url('../assets/img/el_meu_llibre/el-meu-llibre.jpg') no-
repeat fixed center;
  -webkit-background-background-size: cover;
  -moz-background-background-size: cover;
  -o-background-background-size: cover;
  background-size: cover;
}
```

En aquestes línies estem dient que l'etiqueta amb l'identificador **el-meu-llibre** ha de tenir una alçada (**height**) i amplada (**width**) que serà un percentatge de l'amplada i l'alçada de la pantalla i que alhora guardarà uns marges i contindrà una imatge que ocuparà tota la *caixa*.

Si tot ha sortit bé al navegador observarem la Figura 4.



Figura 4. Botó - portada del llibre.

Bé, ara ja podem accedir al nostre llibre, així que comencem a editar-ho.

### Editant el llibre

Si veiem l'aparença del l'enllaç del nostre llibre veiem que ara mateix és una vista en blanc amb un títol i res més. El primer pas serà incloure el menú lateral per a poder navegar a altres zones de l'aplicació.

Per a inserir el menú lateral escriurem les següents línies al fitxer **src/app/el-meu-llibre/el-meu-llibre.page.html** just abans de l'etiqueta **<ion-title>**.

```
<ion-buttons slot="start">
  <ion-menu-button></ion-menu-button>
</ion-buttons>
```

Amb això ja se'ns mostra el menú i podem navegar a les altres vistes de l'aplicació. Podem aprofitar i modificar el títol pel nom del nostre llibre.

A aquesta primera pàgina podem mostrar un altre cop la portada del llibre i la temàtica del mateix per que la gent sàpiga sobre que tracta l'aventura. Per fer això haurem d'introduir el següent text al mateix arxiu que hem modificat amb anterioritat.

```
<ion-card>
  <div id="el-meu-llibre" aria-hidden="true" viewBox="0 0 10 10"></div>
  <p>La razón de la sinrazón, que a mi razón se hace, de tal manera mi
  razón enflaquece, que con razón me quejo de la vuestra fermosura</p>
</ion-card>
```

El següent pas és la modificació de l'arxiu **src/app/el-meu-llibre/el-meu-llibre.page.scss** per que aparegui la imatge de fons i alinear el text de forma adient. A aquest arxiu afegirem les següent línies.

```
#el-meu-llibre{
  margin: 2vh 5vw;
  height: 50vh;
  width: 90vw;

  background: url('../assets/img/el_meu_llibre/el-meu-
llibre.jpg') no-repeat fixed center;
  -webkit-background-size: cover;
  -moz-background-size: cover;
  -o-background-size: cover;
  background-size: cover;
}

p{
  text-align: justify;
  margin: 2vh 10vw;
}
```

Amb aquests estils hauríem d'aconseguir que el navegador ens mostres la Figura 5.



*Figura 5. Vista de la portada del llibre.*

Ja tenim gairebé enlestida l'accés al llibre. Ara podem triar entre dos formes d'accés: mitjançant un botó d'accés o emprant la portada com un botó d'accés al llibre.

Accedint al llibre des de la portada

Per accedir al llibre pitjant sobre la portada utilitzarem el sistema emprat on creem un enllaç que utilitza la portada per a enllaçar amb una pàgina. Així doncs, ens basarem en aquell codi per a generar aquest.

El codi seria de el següent:

```
<a routerLink="/el-meu-llibre/pag01">

  <div id="el-meu-llibre" aria-hidden="true" viewBox="0 0 10
10"></div>
  <p>La razón de la sinrazón, que a mi razón se hace, de tal manera
mi razón enflaquece, que con razón me quejo de la vuestra fermosura.</p>
</a>
```

Si utilitzem aquest codi llavors quan pitgem a qualsevol part de la pantalla obriríem la primera pàgina del llibre.

Accedint al llibre mitjançant un botó

Per utilitzar aquesta opció hem de crear un nou botó per a enllaçar amb la primera pàgina del llibre. Si volem aquesta opció el codi a utilitzar seria el següent:

```
<ion-button expand="block" routerLink="/el-meu-llibre/pag01">Accedir al
llibre</ion-button>
```

Aquest codi l'afegirem a sota del text, si volem que aparegui a pa part inferior o a sobre. En el meu cas l'he afegit a sota i el resultat es mostra a la .

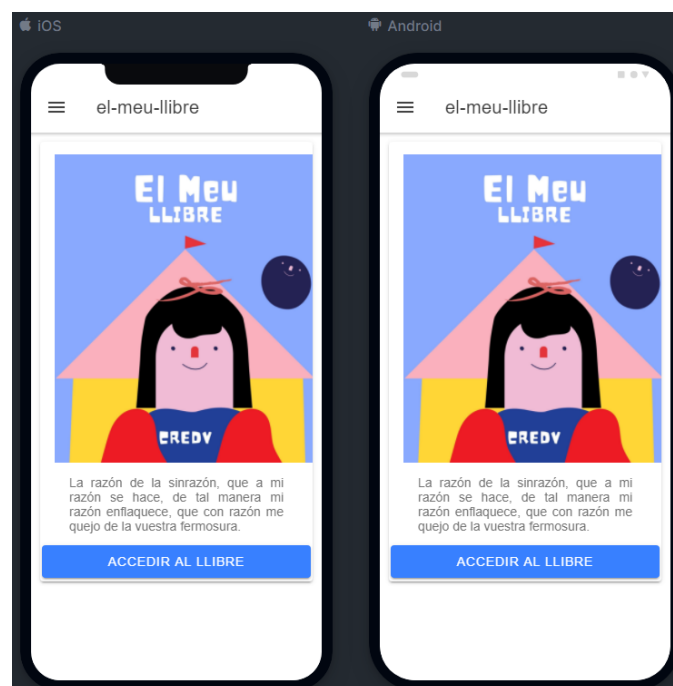


Figura 6. Botó per la portada.

De les dues opcions donades no hi ha una que sigui millor que l'altre; però des del punt de vista de l'accessibilitat i per donar més contingut a la vista es creu que l'opció del botó és millor ja que queda més clar a l'usuari de lector de pantalla i visualment que hem de pitjar al botó per començar a llegir.

Creant una pàgina

Ja hem creat un llibre, ara és el moment de crear la primera pàgina pel nostre llibre. Per fer-ho podem crear un nou component o una nova pàgina. Des del punt de vista pràctic és millor

crear una nova pàgina que un nou component. Per crear una nova pàgina obrirem un terminal o una línia de comandament i escriurem el següent comandament:

---

```
$ionic g page llibres/el-meu-llibre/pag01
```

---

Amb aquest comandament hem creat una nova pàgina. Aquesta pàgina la trobarem a **src/app/el-meu-llibre/**. Ara hem de donar contingut a la primera pàgina del llibre. En aquest manual es donarem 5 plantilles de pàgines diferents. Aquestes son:

- Només text.
- Text amb una imatge.
- Text amb una animació.
- Text amb un vídeo.
- Text amb un so.

#### *Vista amb un text*

La creació d'una vista amb un text i botons de navegació per triar diferents opcions és, possiblement, la opció més senzilla, però alhora la menys vistosa. Per crear aquesta vista obrirem l'arxiu **/src/app/llibres/el-meu-llibre/text/text.page.html** i inserirem el següent codi **HTML** dins l'etiqueta **<ion-content>**:

```
<p>
  Text de la pàgina
</p>
```

En cas de que no es vulgui un títol per la pàgina, esborrarem tot el codi que estui fora de l'etiqueta **<ion-content>**.

Amb el text inclòs a la pàgina, ara només hem de crear els diferents botons per les opcions, un botó per opció. Aquest codi tindrà la següent forma i s'inserirà fora de la etiqueta **<p>**:

```
<ion-button id="opcio-1" shape="round" routerLink="/el-meu-llibre/imatge">Opció 1</ion-button>
<ion-button id="opcio-2" shape="round" routerLink="/el-meu-llibre/animacio">Opció 2</ion-button>
```

Si obrim el navegador veurem la vista de la pàgina 1. L'estil visual de la pàgina no és molt bona. Per tant, hem d'anar a la pàgina d'estils i fer algunes modificacions. L'arxiu d'estil que modificarem és: **src/app/llibres/el-meu-llibre/text/text.page.scss**.

```
p{
  margin: 2vh 5vw;
  text-align: justify;
}

#opcio-1{
  margin: 2vh 7vw;
}
```

```
#opcio-2{
  margin: 2vh 7vw;
}
```

Ara només ens manca incorporar el menú desplegable per a poder accedir a altres parts de l'aplicació. Per fer-ho tornarem a l'arxiu **src/app/llobres/el-meu-llobre/text/text.page.html** i inclourem el següent text.

```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-menu-button></ion-menu-button>
    </ion-buttons>
  </ion-toolbar>
</ion-header>
```

*Vista amb una imatge*

El aquest cas anem a crear una vista amb un text i una imatge. Per fer-ho crearem una nova pàgina utilitzant el terminal o la línia de comandament amb el següent comandament:

---

```
$ionic g page pages/el-meu-llobre/imatge
```

---

Amb això hem creat una nova pàgina dins la carpeta **el-meu-llobre**. Com sempre, obrirem l'arxiu **src/app/llobres/el-meu-llobre/imatge/imatge.page.html** per modificar la part visual de la vista.

Aquesta vista crearem la plantilla per a una vista amb un text i una imatge. Per tant, el que farem serà copiar tot el que hem escrit a l'arxiu **src/app/llobres/el-meu-llobre/text/text.page.html** i l'enganxarem a **src/app/llobres/el-meu-llobre/imatge/imatge.page.html**. Un cop fet això afegirem el codi per inserir una imatge a la pagina.

Com sempre, hi ha dues formes d'incloure una imatge, mitjançant l'etiqueta **<img>** o inserint-la com a fons d'una caixa d'**HTML**. Degut a que és una imatge decorativa la segona opció és la millor des del punt de vista de l'accessibilitat de l'aplicació. El codi per fer-ho seria el següent:

```
<div id="imatge"></div>
```

Aquesta línia l'hem d'inserir just a sobre de l'inici del bloc del text, abans de l'etiqueta **<p>**.

El següent pas es el retoc del estils de la vista. Per això, copiarem el contingut a l'arxiu **src/app/llobres/el-meu-llobre/pag01/pag01.page.scss** i l'enganxarem a l'arxiu **src/app/llobres/el-meu-llobre/imatge/imatge.page.scss**. Amb això ja hem fet més de la meitat de la feina.

Per mostrar la imatge com a fons de la caixa inclourem el següent codi al final de l'arxiu:

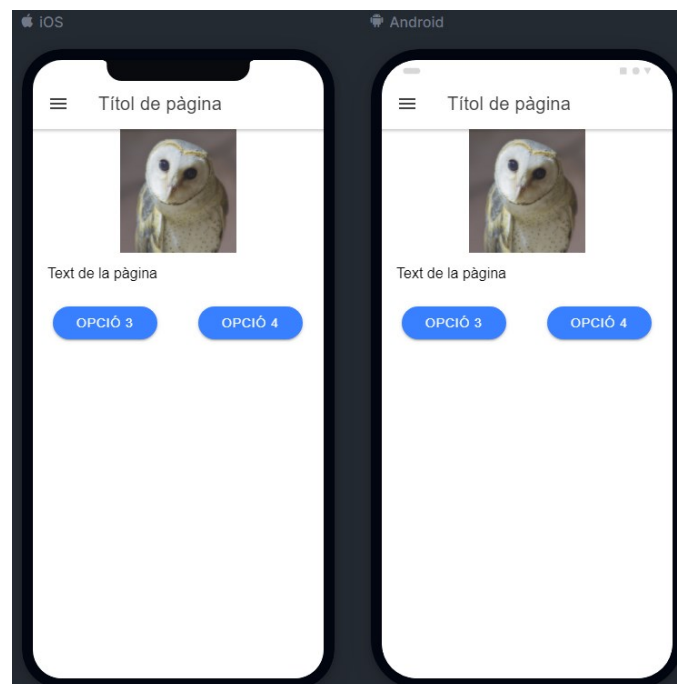
```
#imatge{
  height: 20vh;
  width: 40vw;
```



```
margin: auto;

background: url('../assets/img/el_meullibre/pag02.jpg') fixed
center no-repeat;
-webkit-background-size: cover;
-moz-background-size: cover;
-o-background-size: cover;
background-size: cover;
}
```

Si tot ho he fet de forma correcta la vista tindrà el següent aspecte.



*Figura 7. Vista de text amb imatge.*

#### *Vista amb una animació*

El aquest cas anem a crear una vista amb un text i una animació. Aquesta vista és molt semblant, per no dir idèntica a la vista d'un text amb una imatge; ja que una animació no deixa de ser una imatge amb un format diferent. Per fer-ho crearem una nova pàgina utilitzant el terminal o la línia de comandament amb el següent comandament:

---

```
$ionic g page pages/el-meu-llibre/animacio
```

---

Amb això hem creat una nova pàgina dins la carpeta **el-meu-llibre**. Com sempre, obrirem l'arxiu **src/app/llobres/el-meu-llibre/animacio/animacio.page.html** per modificar la part visual de la vista.

Al ser una vista gairebé idèntica a l'anterior, el que farem serà copiar tot el que hem escrit a l'arxiu **src/app/llobres/el-meu-llibre/imatge/imatge.page.html** i l'enganxarem a

**src/app/llobres/el-meu-llibre/animacio/animacio.page.html**. Un cop fet realitzarem una petita modificació a la etiqueta **<div>**, on el text **id="imatge"** passarà a ser **id="animacio"**.

```
<div id="animacio"></div>
```

El següent pas es el retoc dels estils de la vista. Per això, copiarem el contingut a l'arxiu **src/app/llobres/el-meu-llibre/imatge/imatge.page.scss** i l'enganxarem a l'arxiu **src/app/llobres/el-meu-llibre/animacio/animacio.page.scss**. Amb això ja hem fet pràcticament tot, i només ens caldrà fer un petit retoc, la modificació de la imatge de fons per una animació de fons.

```
#animacio{
  height: 20vh;
  width: 40vw;

  margin: auto;

  background: url('../assets/img/el_meu_llibre/animacio.jpg')
  fixed center no-repeat;
  -webkit-background-size: cover;
  -moz-background-size: cover;
  -o-background-size: cover;
  background-size: cover;
}
```

Si tot ho he fet de forma correcta la vista tindrà el següent aspecte.

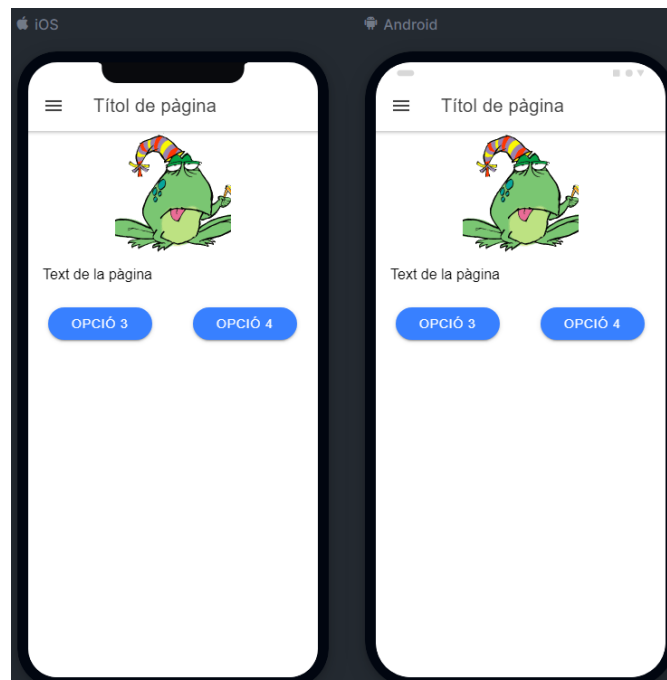


Figura 8. Animació de l'aplicació.

### Vista amb un vídeo

La següent plantilla que generarem serà una amb un text i un vídeo. En aquest cas el més senzill és inserir un vídeo directament d'una pàgina de vídeos como podria ser [Youtube](https://www.youtube.com).

També es pot emmagatzemar un vídeo al dispositiu i que es reproduïxi directament, però tenim el problema de l'espai i de l'augment de memòria que suposa emmagatzemar un vídeo al dispositiu. Encara que d'aquesta manera ens assegurem que l'usuari podrà reproduir sempre el vídeo al no dependre de la seva connexió a internet.

Per crear la pàgina, com sempre, anirem al terminal o a la línia de comandament i escriurem:

---

***\$ionic g page llibre/el-meu-llibre/video***

---

Com sempre, això ens crearà una carpeta dins de **el-meu-llibre**. El següent pas serà anar a [Youtube](#) i triarem un vídeo adient per la pàgina que podem haver pujat nosaltres a la nostra conta o enllaçar des de una conta d'un tercer<sup>2</sup>. Un cop triat el vídeo l'aturarem i triarem l'opció compartir dins de Youtube.



La razón de la sinrazón...

5741 visualizaciones • 21 abr 2016

53 NO ME GUSTA **COMPARTIR** + GUARDAR ...

*Figura 9. Botó de Compartir a Youtube.*

Al fer-ho se'ns obrirà una petita finestra on haurem de triar l'opció **Insertar**.

---

<sup>2</sup>Hem d'assegurar-nos dels permisos dels vídeos de tercers.

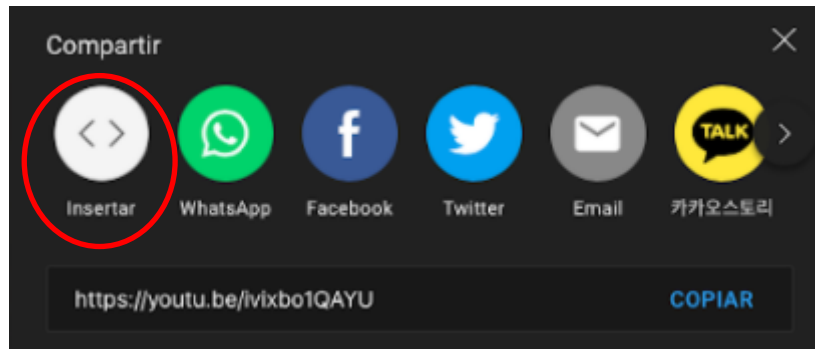


Figura 10. Finestra d'opcions de compartir.

Al pitjar sobre el botó ens apareixerà el codi per a la inserció a la dreta de la finestra. Aquest és el codi que haurem de copiar per després enganxar a l'aplicació.

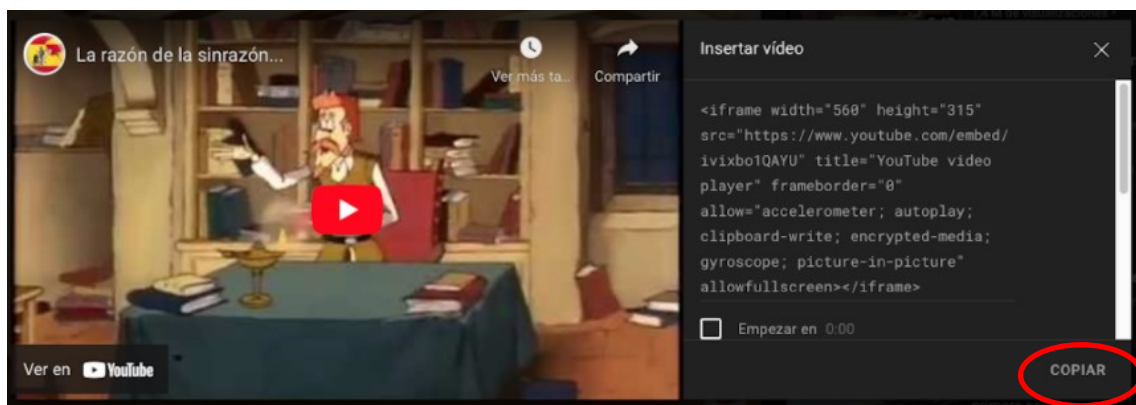


Figura 11. Codi per inserir un vídeo.

Un cop hem copiat el codi per inserir el vídeo de Youtube, obrim l'arxiu **src/app/lilibres/el-meu-llibre/video/video.page.html** i l'inserim just a sota de l'inici de l'etiqueta **<ion-content>**.

```
<ion-content>
  <iframe width="560" height="315"
src="https://www.youtube.com/embed/ivixbo1QAYU" title="YouTube video
player" frameborder="0" allow="accelerometer; autoplay; clipboard-write;
encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
```

Si obrim el navegador per veure com queda el vídeo, veurem que és més gran que la nostra pantalla del navegador, per corregir aquest problema esborrarem les propietats **width** i **height**. D'aquesta forma el vídeo s'adaptarà a la nostra pantalla. També es recomanable modificar la propietat **title** i assignar-li un text més adient pel vídeo<sup>3</sup>.

El següent pas serà obrir l'arxiu **src/app/lilibres/el-meu-llibre/text/text.page.html** i copiar el contingut del **<ion-header>**, així com **<p>** i **<ion-button>** per enganxar-ho a l'arxiu actual. Haurem de fer el mateix amb l'arxiu **src/app/lilibres/el-meu-llibre/text/text.page.scss** on seleccionarem tot el contingut.

Un cop fet tot, el resultat de la vista hauria de ser el següent.

<sup>3</sup> Aquest text el llegiran els lectors de pantalla.

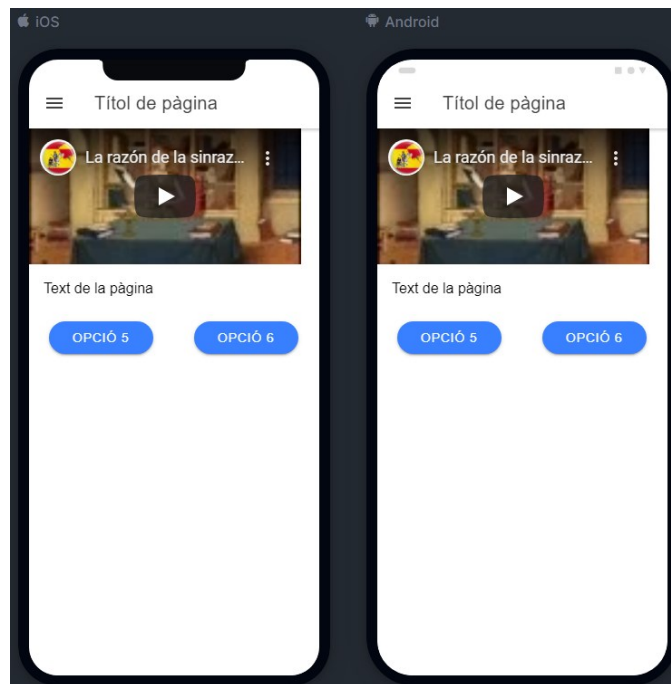


Figura 12. Vídeo a l'aplicació.

#### Vista amb so

La cinquena i darrera plantilla que crearem és una plantilla on quan entrem o al pitjar un botó es llanci un so ambiental que estigui relacionat amb el text de la vista. Per fer-ho crearem una nova pàgina amb el nom de so des de un terminal o una línia de comandaments.

---

***\$ionic g page llibres/el-meu-llibre/so***

---

Com sempre, el primer que farem serà copiar el contingut al fitxers **src/app/llibres/el-meu-llibre/text/text.page.html** i **src/app/llibres/el-meu-llibre/text/text.page.scss** i enganxar el seu contingut als fitxers **src/app/llibres/el-meu-llibre/video/video.page.html** i **src/app/llibres/el-meu-llibre/video/video.page.scss** respectivament. Un cop fet això afegirem les següents línies.

```
<audio autoplay>
  <source src="../../../../assets/sounds/sound-here.mp3"
type="audio/mp3">
  El teu navegador no suporta aquest arxiu de so.
</audio>
```

Si ara anem al navegador i iniciem l'aplicació sentirem un so quan entrem a la pàgina so. S'ha programat que el so es llenci només carregar-se la vista mitjançant l'atribut **autoplay**.

Amb això tindríem totes les plantilles llestes per a ser utilitzades. Al següent capítol s'explica com crear un llibre de zero utilitzant les plantilles que acabem de crear.

## Creant un llibre amb les plantilles

### Com crear un llibre nou

El primer pas crear un llibre utilitzant les plantilles serà crear un nou llibre. Això ho farem des d'un terminal o una línia de comandaments mitjançant el següent comandament dins de la carpeta del nostre projecte Book.

---

```
$ionic g page llibres/nou-llibre
```

---

Amb això crearem una carpeta dins de la carpeta **llibres**. Aquesta carpeta contindrà totes les pàgines del nostre nou llibre.

### Afegint el nou llibre a l'aplicació

Per afegir el nou llibre que acabem de crear a la presentació el que haurem de fer és obrir el fitxer `src/app/folder/folder.page.html` i afegir les següents línies.

```
<button (click)="openBook(0)" data-message="Obrir nou-llibre"
type="button" class="btn">
  <div id="nou-llibre" aria-hidden="true" viewBox="0 0 10 10"></div>
</button>
```

Per incloure un altre títol modificarem el text **nou-llibre** pel títol del llibre. Un cop modificar aquest arxiu obrirem el fitxer `src/app/folder/folder.page.scss` i afegirem les següents línies.

```
#nou-llibre{
  margin: 2vh 10vw;
  height: 40vh;
  width: 80vw;

  background: url('../assets/img/nou_llibre/nou-llibre.jpg') no-repeat
fixed center;
-webkit-background-background-size: cover;
-moz-background-background-size: cover;
-o-background-background-size: cover;
background-size: cover;
}
```

En aquest cas farem les següent modificacions:

- Modificarem el text **nou-llibre** pel títol de llibre.
- Modificarem la propietat **url** de **background**, on indicarem la ruta de la imatge que ens servirà de portada.

Si tot s'ha fet bé, al iniciar l'aplicació en un navegador ens sortirà la portada del nou llibre en pantalla. Ara només manca que al pitjar sobre el llibre se'ns obri el mateix.

Per fer això obrirem el fitxer `src/app/folder/folder.page.ts` i afegirem el nou llibre a l'array amb els títols del llibres que ja tenim a l'aplicació. Recordem que els títol s'han de separar per una , (coma) i escriure's entre " " (cometes dobles, com a l'exemple.

```
llibres:string=["/el-meu-llibre", "/nou-llibre"];
```

Si s'ha fet de forma correcta al pitjar sobre la portada del llibre se'ns mostrarà la primera pàgina del llibre en blanc.

### Creant la portada

Un cop afegir el llibre a l'aplicació hem de crear la seva portada interior i el botó per iniciar la lectura. Per això obrirem el fitxer **src/app/llibres/nou-llibre/nou-llibre.page.html** i inclourem el text per que ens mostri la portada i el botó per iniciar la lectura. Per fer això haurem d'incloure les següents línies.

```
<ion-card>
  <div id="nou-llibre" aria-hidden="true" viewBox="0 0 10 10"></div>
  <p>La razón de la sinrazón, que a mi razón se hace, de tal manera mi
  razón enflaquece, que con razón me quejo de la vuestra fermosura.</p>

  <ion-button expand="block" routerLink="/nou-llibre/text">Accedir al
  llibre</ion-button>
</ion-card>
```

En aquestes línies haurem de realitzar les següents modificacions:

- **nou-llibre**. Ho substituïrem pel títol del nou llibre.
- **<p>**. El contingut a entre **<p>** i **</p>** s'ha de modificar amb un text adient pel nou llibre.
- **routerLink**. A l'atribut **routerLink** escriurem el nom del llibre on posa **nou-llibre** i on posa **text** escriurem el nom de la primera pàgina.

Per que ens aparegui la portada del llibre haurem de realitzar modificacions al fitxer **src/app/llibres/nou-llibre/nou-llibre.page.scc**. Aquí copiarem les següents línies:

```
#nou-llibre{
  height: 50vh;
  width: 90vw;

  margin: auto;

  background: url('../assets/img/nou_llibre/nou-llibre.jpg') no-
  repeat fixed center;
  -webkit-background-background-size: cover;
  -moz-background-background-size: cover;
  -o-background-background-size: cover;
  background-size: cover;
}

p{
  text-align: justify;
  margin: 2vh 10vw;
}
```

Haurem de modificar **nou-llibre** amb el títol del nostre llibre, i a la propietat **url** de **background** haurem d'escriure la direcció del dibuix que utilitzarem com a portada pel llibre.

Ara mateix al nostre navegador hauríem de tenir la portada del llibre i un botó per iniciar la lectura del mateix.

## Creant una pàgina

Per crear una nova pàgina haurem d'anar al terminal o línia de comandaments i escriure el següent comandament:

---

```
$ionic g page llibres/nou-llibre/nom-pagina
```

---

En lloc de anomenar les pàgines com a **pag01**, que és una possibilitat, es recomana assignar un nom (títol) a cada pàgina on es faci menció a la opció triada per l'usuari. Per exemple, si hem creat un botó amb el text **CONTINUAR PERSECUCIÓ**, el nom de la pàgina podria ser **continuar-persecucio**. D'aquesta manera ens serà més fàcil relacionar les pàgines amb els seus enllaços.

### Com afegir un text

Per afegir un text a la nostra pàgina copiarem el contingut dels fitxers **src/app/llibres/el-meu-llibre/text/text.page.html** i **src/app/llibres/el-meu-llibre/text/text.page.scss**. Al fitxer **src/app/llibres/el-meu-llibre/text/text.page.scss** no realitzarem cap modificació, el deixarem tal qual i al fitxer **src/app/llibres/el-meu-llibre/text/text.page.html** realitzarem les següents modificacions:

- **<ion-title>**. Assignarem un títol a la pàgina, si volem. En cas de no voler eliminarem l'etiqueta.
- **<p>**. Modificarem el text amb el text que correspongui a la nostra història i a la selecció triada per l'usuari.
- **<ion-button>**. Modificarem el següent:
  - **routerLink**. Escriurem el text entre cometes cobles (" "), l'enllaç a la pàgina següent, segons l'esquema: **/nom-llibre/nom-pagina**.
  - **Opció 1 i Opció 2**. Escriurem un text breu amb l'opció donada al lector, per exemple: **CONTINUAR PERSECUCIÓ**.

### Com afegir una imatge

Per afegir una imatge a la nostra pàgina copiarem el contingut dels fitxers **src/app/llibres/el-meu-llibre/imatge/imatge.page.html** i **src/app/llibres/el-meu-llibre/imatge/imatge.page.scss**. Al fitxer **src/app/llibres/el-meu-llibre/imatge/imatge.page.html** realitzarem les següents modificacions:

- **<ion-title>**. Assignarem un títol a la pàgina, si volem. En cas de no voler eliminarem l'etiqueta.
- **<p>**. Modificarem el text amb el text que correspongui a la nostra història i a la selecció triada per l'usuari.
- **<ion-button>**. Modificarem el següent:
  - **routerLink**. Escriurem el text entre cometes cobles (" "), l'enllaç a la pàgina següent, segons l'esquema: **/nom-llibre/nom-pagina**.
  - **Opció 3 i Opció 4**. Escriurem un text breu amb l'opció donada al lector, per exemple: **CONTINUAR PERSECUCIÓ**.



Al fitxer **src/app/llobres/el-meu-llibre/imatge/imatge.page.scss** haurem de modificar la imatge. Per això modificarem a l'atribut **background** la propietat **url**, on escriurem la direcció de la carpeta on tenim emmagatzemada la imatge.

Si hem guardat la imatge a una carpeta amb el nom del llibre i dins de la carpeta **img** que s'ha creat per fer l'aplicació, només haurem de modificar el següent text: **el-meu-llibre/imatge** per **nom-del-libre/nom-de-la-imatge**. Recordem que també s'ha de descriure el tipus d'imatge: **.jpg**, **.png**, **.tiff**, ...

### Com afegir una animació

El procés per afegir una animació és pràcticament idèntic al d'afegir una imatge, doncs una animació no deixa de ser un tipus especial d'imatge.

Per afegir una animació a la nostra pàgina copiarem el contingut dels fitxers **src/app/llobres/el-meu-llibre/animacio/animacio.page.html** i **src/app/llobres/el-meu-llibre/animacio/animacio.page.scss**. Al fitxer **src/app/llobres/el-meu-llibre/animacio/animacio.page.html** realitzarem les següents modificacions:

- **<ion-title>**. Assignarem un títol a la pàgina, si volem. En cas de no voler eliminar l'etiqueta.
- **<p>**. Modificarem el text amb el text que correspongui a la nostra història i a la selecció triada per l'usuari.
- **<ion-button>**. Modificarem el següent:
  - **routerLink**. Escriurem el text entre cometes cobles (" "), l'enllaç a la pàgina següent, segons l'esquema: **/nom-libre/nom-pàgina**.
  - **Opció 5** i **Opció 6**. Escriurem un text breu amb l'opció donada al lector, per exemple: **CONTINUAR PERSECUCIÓ**.

Al fitxer **src/app/llobres/el-meu-llibre/animacio/animacio.page.scss** haurem de modificar la imatge. Per això modificarem a l'atribut **background** la propietat **url**, on escriurem la direcció de la carpeta on tenim emmagatzemada l'animació.

Si hem guardat la imatge a una carpeta amb el nom del llibre i dins de la carpeta **img** que s'ha creat per fer l'aplicació, només haurem de modificar el següent text: **el-meu-llibre/animacio** per **nom-del-libre/nom-de-la-animacio**. Recordem que també s'ha de descriure el tipus d'animació: **.gif**, **.jpeg**, ...

### Com afegir un vídeo

Per afegir un vídeo a la nostra pàgina copiarem el contingut dels fitxers **src/app/llobres/el-meu-llibre/video/video.page.html** i **src/app/llobres/el-meu-llibre/video/video.page.scss**. Al fitxer **src/app/llobres/el-meu-llibre/video/video.page.scss** no realitzarem cap modificació, el deixarem tal qual i al fitxer **src/app/llobres/el-meu-llibre/video/video.page.html** realitzarem les següents modificacions:

- **<ion-title>**. Assignarem un títol a la pàgina, si volem. En cas de no voler eliminar l'etiqueta.
- **<iframe>**. Esborrarem l'etiqueta i enganxarem l'etiqueta amb el mateix nom que hàgim obtingut de **Youtube**, o qualsevol altre servidor de vídeos en streaming. Recordem que per fer el vídeo **responsive** hem d'esborrar els atributs **height** i **width**.

- **<p>**. Modificarem el text amb el text que correspongui a la nostra història i a la selecció triada per l'usuari.
- **<ion-button>**. Modificarem el següent:
  - **routerLink**. Escriurem el text entre cometes cobles (" "), l'enllaç a la pàgina següent, segons l'esquema: **/nom-llibre/nom-pàgina**.
  - **Opció 7 i Opció 8**. Escriurem un text breu amb l'opció donada al lector, per exemple: **CONTINUAR PERSECUCIÓ**.

#### Com afegir un so

Per afegir un so a la nostra pàgina copiarem el contingut dels fitxers **src/app/llobres/el-meu-llibre/so/so.page.html** i **src/app/llobres/el-meu-llibre/so/so.page.scss**. Al fitxer **src/app/llobres/el-meu-llibre/so/so.page.scss** no realitzarem cap modificació, el deixarem tal qual i al fitxer **src/app/llobres/el-meu-llibre/so/so.page.html** realitzarem les següents modificacions:

- **<ion-title>**. Assignarem un títol a la pàgina, si volem. En cas de no voler eliminarem l'etiqueta.
- **<source>**. A l'atribut **src** escriurem l'enllaç a l'arxiu de so. Si guardem els arxius de so a dins de la carpeta **sounds** creada, només hem de modificar el nom de l'arxiu de so, **sound-here** a l'exemple, pel nom del nostre arxiu. Recordem que hem d'afegir el tipus d'arxiu sonor: .wav, .mp3, .mp4, ... Per últim, si no volem que el so es reproduïxi al entrar a la pàgina hem d'eliminar l'opció **autoplay** de l'etiqueta **<audio>**.
- **<p>**. Modificarem el text amb el text que correspongui a la nostra història i a la selecció triada per l'usuari.
- **<ion-button>**. Modificarem el següent:
  - **routerLink**. Escriurem el text entre cometes cobles (" "), l'enllaç a la pàgina següent, segons l'esquema: **/nom-llibre/nom-pàgina**.
  - **Opció 9 i Opció 10**. Escriurem un text breu amb l'opció donada al lector, per exemple: **CONTINUAR PERSECUCIÓ**.