

---

# **NewPortaL Documentation**

**LFG Scavelli**

**19 ott 2018**



<b>1</b>	<b>User Guide</b>	<b>3</b>
1.1	Overview	3
1.1.1	Requirements	3
1.1.2	Installation	3
1.1.3	License	6
1.1.4	Contributing	6
1.2	Quickstart	6
1.2.1	Front end with widgets	6
1.2.2	Web Content	7
1.2.3	Dynamic Data List	8
1.2.4	Categorization	8
1.2.5	Pages, themes and layouts	8
1.2.6	User management	8
1.3	Widgets principali	8
1.3.1	Webcontent widget	9
1.3.2	ContentList widget	10



NewPortaL è un semplice Content Management System (CMS) basato sui frameworks [Laravel 5.7](#) e [Angular 7](#), creato e mantenuto da LFG Scavelli.

Il presente progetto è open source, sotto licenza MIT, ed è utilizzabile per ogni scopo, personale o commerciale.

Il pacchetto originario NewPortaL, sviluppato interamente in PHP senza l'ausilio di alcun framework, ha consentito la generazione di numerosi portali orizzontali per la Pubblica Amministrazione e la produzione e gestione di servizi specifici rivolti al cittadino.

L'attuale revisione utilizza inoltre i seguenti pacchetti open source: AdminLTE (per il tema di backend); - Startbootstrap-grayscale (per il tema di Frontend); - Eonasdan-bootstrap-datetimepicker; - Formeo; - Ckeditor per l'editor dei testi e Ace per il code Editor e molti altri.

<b>Avvertimento:</b> E' vivamente sconsigliato l'uso in produzione. Utilizzabile ai soli fini di sviluppo.
--



## 1.1 Overview

### 1.1.1 Requirements

E' necessario assicurarsi che il server soddisfi i seguenti requisiti:

- PHP >= 7.1.3
- OpenSSL PHP Extension
- PDO PHP Extension
- Mbstring PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension
- Ctype PHP Extension
- JSON PHP Extension

---

**Nota:** NewPortal si basa su [Laravel v5.7](#), pertanto i requisiti sono quelli richiesti dal framework Laravel

---

### 1.1.2 Installation

Si presume che, sia in locale che su uno spazio hosting Linux con accesso SSH, siano installati i pacchetti Node.js - Npm - PHP - Composer. Una valida alternativa sarebbe l'installazione e l'uso di Homestead, un box Vagrant pre-configurato che fornisce un ambiente di sviluppo completo di tutti i pacchetti necessari.

### Su una macchina locale di sviluppo

```
# importare il progetto da github repository
# posizionarsi nella document root ex cd /c/www
git clone https://github.com/lscavelli/newportal.git

# in alternativa a "git clone" è possibile scaricare l'app dal seguente indirizzo
# https://github.com/lscavelli/newportal/archive/master.zip

# installare le dipendenze di Back end
cd newportal
composer install

# copiare il file di environment
cp .env.example .env

# generare la chiave del portale
php artisan key:generate

# impostare l'accesso al DB all'interno del file .env
# effettuare la generazione delle tabelle nel DB
# attendere qualche minuto, le tabelle Cities e Countries contengono molti dati
php artisan migrate --force --seed

# rendere scrivibile le seguenti dir
chmod -R g+w storage
chmod -R g+w bootstrap/cache

# Avvio il server web
php artisan serve // http://127.0.0.1:8000

# modificare il file hosts da /etc/hosts (linux)
# C:\Windows\System32\drivers\etc\hosts (windows), inserendo:
127.0.0.1 newportal.test

# installare le dipendenze di front end
cd public
npm install

# credenziali per login - http://newportal.test:8000/login
username: admin@example.com
password: admin
```

### Su uno spazio hosting Linux con accesso SSH

```
# nomesito.io (Domain name)
# httpdocs (Document root)

# verificare che la dir httpdocs sia vuota, dopodiché eliminarla
rm -rf httpdocs

# se già esiste la dir newportal rimuoverla con rm -rf newportal
# importare il progetto laravel fuori dalla document root
git clone https://github.com/lscavelli/newportal.git

# in alternativa a "git clone" è possibile scaricare l'app dal seguente indirizzo
```

(continues on next page)

(continua dalla pagina precedente)

```
# https://github.com/lscavelli/newportal/archive/master.zip

# creare un link simbolico alla dir public
# dopo la creazione del link verificare la funzionalità, entrando nella cartella - cd
->httpdocs
ln -s newportal/public httpdocs

# installare le dipendenze di Back end
cd newportal
composer install

# copiare il file di environment
cp .env.example .env

# generare la chiave del portale
php artisan key:generate

# eliminare la linea APP_ENV dal file .env (questo verrà impostato su production)
# eliminare la linea APP_DEBUG dal file .env (questo verrà impostato su false)
# impostare nel file .env la variabile SESSION_DRIVER su database

# impostare i dati di accesso al DB nel file .env
# effettuare la generazione delle tabelle nel DB
# Attendere qualche minuto. Le tabelle Cities e Countries contengono molti dati
php artisan migrate --seed

# verificare che i permessi per le cartelle e i file siano impostati correttamente
#cd /var/www
#chown -R <utente-ftp>:<apache> newportal
#find newportal -type f -exec chmod 644 {} \;
#find newportal -type d -exec chmod 755 {} \;

# abilitare la scrittura per alcune dir
chmod -R o+w storage
chmod -R o+w bootstrap/cache

# ottimizzare l'autoloader e metto in cache alcuni file
composer dumpautoload -o // oppure con composer dump-autoload --optimize --no-dev
php artisan config:cache
php artisan route:cache

# una volta che si esegue il comando config:cache.
# due nuovi file saranno creati in bootstrap/cache.
# Questi sono config.php e services.php
# Rieseguire il comando se si cambiano i percorsi e le configurazioni
# Con il comando route:cache viene creato nella cache un terzo file, route.php.

# Installo le dipendenze di front end
cd httpdocs
npm install

# credenziali per login - http://<domain-name>/login
username: admin@example.com
password: admin
```

### 1.1.3 License

Licensed using the [MIT license](#).

Copyright (c) 2017 LFG Scavelli <<https://github.com/lscavelli>>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the «Software»), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED «AS IS», WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.1.4 Contributing

Per contribuire al progetto si dovrà installare in locale newportal seguendo la *procedura di installer* indicata nella presente guida. Si dovranno installare anche tutte le dipendenze di backend e frontend, utilizzando rispettivamente Composer e npm. Al termine dei lavori di integrazione/modifica sarà sufficiente aprire una pull request su GitHub

Gli errori, ma anche semplici suggerimenti, potranno essere segnalati attraverso il webform di contatto del sito web o l'email [newportalclub@gmail.com](mailto:newportalclub@gmail.com).

Sarebbe auspicabile che le vulnerabilità di protezione venissero segnalate con urgenza esclusivamente tramite indirizzo di posta elettronica [newportalclub@gmail.com](mailto:newportalclub@gmail.com)

E' possibile contribuire al progetto anche in ambiti diversi dallo sviluppo software, ovvero nella traduzione dei testi, nell'analisi, nell'eseguire test di unità, nel design etc...

Ogni contributo risulterà sempre prezioso e utile al miglioramento del progetto.

## 1.2 Quickstart

Questa pagina fornisce una rapida introduzione alle principali caratteristiche di Newportal. Se non hai ancora installato **Newportal** vai alla pagina *Installation*.

### 1.2.1 Front end with widgets

Un **widget** è un componente modulare che aggrega contenuti informativi su una pagina web. Una sorta di plug-in che consente la pubblicazione di informazioni in porzioni di pagina. Tramite i widgets lo stesso contenuto è utilizzabile in più pagine o più volte nella stessa pagina (Multi istanza). I widgets possono essere trascinate in aree specifiche della pagina in base al modello di layout adottato. Durante il trascinamento si attiva un indicatore di colore blue in corrispondenza delle aree di rilascio. All'interno della stessa area sarà possibile disporre più widgets e ordinarle diversamente. Grazie al supporto al Drag-and-Drop, le pagine del portale potranno essere riorganizzate rapidamente.

## 1.2.2 Web Content

I contenuti web sono contenuti strutturati che compongono le pagine del portale. Si basano essenzialmente sulle Strutture (form per l'acquisizione dei dati informativi) e sui Modelli (Viste di presentazione dei dati). Un tool visuale, basato su [Formeo](#), consente agevolmente di costruire, in base alle proprie esigenze, strutture semplici e complesse alle quali sarà possibile associare uno o più modelli (dynamic template). Lato front-end i widgets possono estrarre Web Content che rispondano a determinate strutture (ex. news / eventi) ed assumere differenti aspetti in base al modello scelto per la visualizzazione.

Per ogni struttura `/admin/structure` è possibile creare più modelli. Di seguito si riportano alcuni esempi di viste dinamiche associate alle strutture formeo.

Esempio di modello di tipo lista associato ad una struttura contenente un solo campo variabile: es. `$np714bc8ec8c2a4929bee9319dd31a1207`

```

1 <li class="articles clearfix content_divisor">
2   <a href="{!! $np_href !!}" class="content_title">{!! $np_title !!}</a><br />
3   <span class="image left"></span>
4   <p>{!! \App\Libraries\sl_text::sommario(strip_tags(
↳$np714bc8ec8c2a4929bee9319dd31a1207)) !!>
5     @if(strlen(strip_tags($np714bc8ec8c2a4929bee9319dd31a1207))>255)
6       <a href="{!! $np_href !!}">Leggi ancora</a>
7     @endif
8   </p>
9   <p class="articles_meta">
10     @if(count($np_categories)>0)
11       @foreach($np_categories as $category)
12         <a class="articles_category" href="/{!! $np_page !!}?category={!!
↳$category->id !!}"#{!! $category->name !!}</a>,
13       @endforeach
14     @endif
15     Posted by <a href="/{!! $np_page !!}?author={!! $np_author_id !!}" class=
↳"articles_author">{!! $np_author_name !!}</a>
16     on <time datetime="{!! $np_data_creazione !!}">{!! $np_data_creazione !!</
↳time>
17   </p><br />
18 </li>

```

Oltre all'autore, all'immagine e al titolo, la precedente vista visualizza anche le categorie associate ai singoli contenuti web. Allo stesso modo potrebbe visualizzare i Tag.

Il successivo esempio riporta una lista semplificata contenente solo titolo e data di creazione.

```

1 <li class="block-hover">
2   <a href="{!! $np_href !!}" class="content_title">{!! $np_title !!}</a>
3   <div style="font-size:9px">{!! $np_data_creazione !!}</div>
4 </li>

```

Nel successivo esempio, invece, si fa riferimento ad una vista relativa ad un singolo contenuto web.

```

1 <h3>{!! $np_title !!}</h3>
2 by <a href="/video?author={!! $np_author_id !!}" class="articles_author">{!! $np_
↳author_name !!}</a>
3 on {!! $np_data_creazione !!>
4 <br /><br />
5 {!! $np714bc8ec8c2a4929bee9319dd31a1207 !!>

```

### 1.2.3 Dynamic Data List

Attraverso i Dynamic Data List è possibile definire un elenco dati dinamico, ovvero un elenco di contenuti web basato sulla medesima struttura, esportabile nei diversi formati (excel, csv, xml, txt, etc. . . ). Così come avviene per i Contenuti web, la creazione dei «Dynamic Data List» richiede l'impostazione di una struttura dati.

### 1.2.4 Categorization

Newportal consente la creazione di più vocabolari per aggregare un numero quasi infinito di categorie omogenee. I vocabolari possono essere associati a più servizi del portale e resi accessibili separatamente attraverso la sezione «Categorizzazione». Contestualmente alle categorie è possibile definire e associare i tag, in modo da rendere i contenuti facilmente ritrovabili e filtrabili anche lato front-end.

#### Frontend side

Con l'uso di appositi widgets sarà possibile creare pagine web complete di menu di navigazione basati su vocabolari di categorie e tags. Gli elementi che aggregano i contenuti presenti nella pagina consentiranno di filtrare i dati in base ai valori dei vocabolari definiti in configurazione o in base alla categoria passata nell'url, se la comunicazione tra widgets è stata abilitata.

### 1.2.5 Pages, themes and layouts

Newportal gestisce sia pagine pubbliche (accessibili a chiunque) che private. Le pagine private sono accessibili solo agli utenti membri del sito che possiede le pagine. Per impostazione predefinita tutte le pagine vengono automaticamente mostrate nei menu di navigazione. Se si vuole omettere tale visualizzazione, la pagina dovrà essere creata come nascosta. Le pagine possono essere: - nidificate ovvero create come sotto-pagine di altre pagine esistenti; - vuote o precompilate, partendo da dei modelli di pagina predefiniti; - di tipo Widget, URL o Link interno. In base al tema e al layout scelto, le pagine possono assumere aspetti differenti. I layout consentono di specificare come i widgets saranno disposti nelle pagine. Possono essere creati e resi disponibili per lo stesso sito diversi layout. Con alcune piccole modifiche sarà possibile utilizzare themes e templates fully responsive facilmente reperibili su web

### 1.2.6 User management

Dal pannello di controllo un amministratore può: gestire gli utenti del portale, raggruppandoli in team di lavoro o inserendoli in organizzazioni gerarchiche; assegnare ad utenti e gruppi di utenti i permessi (privilegi di accesso alle risorse del sistema) o i ruoli (insieme di permessi); accedere a tutti i profili definiti dal sistema (utente, gruppo, permesso, ruolo, organizzazione). Se abilitato, il modulo di registrazione per creare i nuovi utenti può essere liberamente accessibile. Attraverso la pagina di login gli utenti potranno autenticarsi al portale, digitando il proprio nome utente (o email) e password, oppure utilizzando alcune delle opzioni sociali di autenticazione. Qualsiasi accesso alle risorse protette del sistema effettuato da un utente non registrato sarà rediretto alla pagina di login. L'attività degli utenti è registrata. Un utente non amministratore avrà accesso solo al registro della propria attività. Il sistema consente anche la visualizzazione delle sessioni attive con possibilità per l'utente amministratore di cancellarle.

## 1.3 Widgets principali

Nella pagina Quikstart si è fatto cenno ai *Widgets* come elementi aggregatori di contenuti informativi su una pagina web Newportal. I widgets possono essere trascinate (in modalità Drag-and-drop) in aree specifiche delle pagine (Dropped aree) in base ai modelli di layout adottati. I widget attualmente disponibili sono:

- *webcontent*;
- *contentlist*;
- *navigationpages*;
- *navigationcat*;
- *navigationtag*;
- *organizations*.

Altri widgets sono in fase di realizzazione.

### 1.3.1 Webcontent widget

Il widget «WebContent» consente la visualizzazione di un singolo contenuto informativo nella posizione in cui è posizionato il widget stesso. L'aspetto del contenuto dipenderà dalla sua struttura dati e dal modello (vista) scelto per la sua visualizzazione. Lo stesso contenuto informativo, attraverso l'uso di più widgets WebContent, può essere presente in più pagine del sito o più volte nella stessa pagina (in modalità multi istanza), assumendo, in base ai modelli scelti, differenti aspetti.

La scelta del modello può essere effettuata sia nell'editor del contenuto, sia nella finestra di configurazione del widget.

I modelli possono essere di due tipi: «lista», quelli utilizzabili nel widget «contentList» esclusivamente per le righe di una lista; «base», applicabili al singolo contenuto informativo ed utilizzabili nel widget «webContent». I modelli che verranno mostrati nella finestra di configurazione del widget, dipenderanno dalla struttura dati del contenuto selezionato.

Poiché possono essere impostati, per lo stesso contenuto, più modelli, il widget in fase di visualizzazione del contenuto informativo utilizzerà il seguente ordine di priorità:

1. Modello impostato nel widget;
2. Modello impostato nell'editor del contenuto;
3. In assenza dei primi due, il primo modello appartenente alla struttura dati del contenuto.

Se il modello non è trovato o non esistono modelli per la struttura del contenuto selezionato, il sistema mostrerà un messaggio di errore.

#### Selezione manuale del contenuto

Il contenuto informativo da visualizzare nel widget, viene selezionato, manualmente, nel pannello di configurazione del widget. Dopo aver scelto una struttura dati, la finestra mostrerà i contenuti informativi basati sulla struttura selezionata. Cliccare su *aggiungi*, per selezionare il contenuto.

#### Selezione dinamica del contenuto

Se la comunicazione del widget è attiva (impostabile dalla finestra di configurazione) e nell'URL è presente lo slug di un contenuto informativo valido, questo sarà ricercato e visualizzato nel widget, indipendentemente dalla presenza di un contenuto impostato manualmente nel widget. Con tale configurazione un widget webContent potrebbe visualizzare di default un contenuto informativo e dinamicamente un altro. Nella modalità dinamica lo stesso contenuto informativo verrà visualizzato in tutti i widgets webContent che avranno la comunicazione impostata su «attiva».

Uno slug di un contenuto informativo può essere presente nell'url nei seguenti modi:

- `http://<domain-name>/<page>/<slug-web-content>`
- `http://<domain-name>?content=<slug-web-content>`

- `http://<domain-name>/<slug-web-content>`

Quest'ultima ipotesi sarà attuabile solo e soltanto se risulterà assegnata al contenuto informativo una pagina di visualizzazione (Display Page) contenente un widget di tipo `webContent` con il parametro «predefinito per questa pagina» impostato su «On».

**Avvertimento:** La visualizzazione dinamica è consentita solo se i contenuti passati nell'url risultano pubblici (Modalità predefinita) ossia non salvati con l'opzione «gestibili manualmente». Nel caso contrario i contenuti dovranno essere selezionati manualmente attraverso il pannello di configurazione del widget.

### Condivisione del contenuto

Dal pannello di configurazione è possibile abilitare la condivisione del contenuto informativo sui principali social media. Se la condivisione è abilitata, compariranno, a fine testo, sulla destra, le icone dei social disponibili. Cliccando su queste icone si aprirà un popup che consentirà lo sharing del content web sui rispettivi social network. Se al contenuto informativo è stata associata un'immagine, questa comparirà nella finestra di condivisione del social network facebook.

### Abilitazione dei commenti

Il widget `webContent` consente l'attivazione del modulo per l'invio e la visualizzazione, a piè di pagina, dei commenti. I commenti, in base alla configurazione scelta, possono essere moderati o immediatamente visibili. All'atto della registrazione di un nuovo commento, una notifica di email verrà inviata all'autore del contenuto. I commenti possono essere gestiti nella sezione dei contenuti, dal menu posto a destra in corrispondenza del contenuto interessato.

### Abilitazione del conteggio delle visite

Dal pannello di configurazione del widget è possibile attivare il conteggio delle visite al contenuto informativo. Il numero complessivo delle visite può essere visualizzato nella sezione web content del cruscotto di amministrazione, in corrispondenza del contenuto interessato. Il dato sarà considerato dal widget `contentList` per visualizzare la lista dei contenuti più popolari.

### Abilitazione dell'evidenziazione della sintassi

Spesso i contenuti informativi includono, al proprio interno, dei frammenti di codice. Per facilitare la lettura di questo particolare testo, il widget «`webContent`» consente l'abilitazione dell'evidenziazione della sintassi - `syntax highlighting`. Un sistema che supporta una vasta gamma di linguaggi di programmazione, di markup e di scripting ed effettua la colorazione della sintassi in maniera completamente automatica.

### 1.3.2 ContentList widget

Il widget `contentList` consente di visualizzare una lista di contenuti formattata in base a determinati modelli scelti nel pannello di configurazione del widget. Lo stile del contenitore esterno della lista è preimpostato sul modello «`listAsset`», ma può essere comodamente sostituito con altri modelli, anche custom, presenti nella cartella «`view`» del widget «`contentList`». In taluni casi si potrebbe aver bisogno di template generici legati espressamente al tema corrente. Questi ultimi sono raccolti nella cartella «`partials`» del tema corrente e possono essere selezionati dalla scheda «Altre impostazioni» del pannello di configurazione del widget.

Attraverso questo widget la stessa lista può essere utilizzata in più pagine o più volte nella stessa pagina (Multi istanza), assumendo, in base ai modelli scelti, aspetti differenti.

**Avvertimento:** Nella lista saranno presenti solo i contenuti pubblici ossia quelli che non risultano salvati con l'opzione «gestibili manualmente».

### Filtraggio in base ai Tags e Categorie

I contenuti della lista possono essere filtrati in base ai tags e categorie definiti nel sito. Per l'impostazione dei filtri il widget utilizzerà il seguente ordine di priorità:

1. Parametri `tag` e `category` impostati nell'url (Se la comunicazione del widget è attiva);
2. Tags e categories del contenuto passato nella URL tramite slug (Se la comunicazione del widget è attiva);
3. Tags e categories impostati dal pannello di configurazione del widget (Se la comunicazione è disattiva o i punti precedenti non producono un risultato).

### Filtraggio in base alle strutture dati

Il widget `contentList` consente di filtrare i contenuti informativi in base ad una particolare struttura dati definita nel pannello di configurazione del widget. Non supporta la modalità dinamica.

### Filtraggio in base all'autore del contenuto

Passando semplicemente nell'url il parametro «author» con l'identificativo dell'autore, i contenuti informativi saranno filtrati in base all'autore. Questo filtro si aggiungerà agli altri filtri.

### Visualizzazione dei contenuti più popolari

I contenuti informativi possono essere ordinati anche in base al numero delle visite. Se la direzione è discendente, in testa comparirà il contenuto con il maggior numero di visite, altrimenti il contenuto con il minor numero di visite.

### Ordinamento delle voci

Oltre che per «visite», i contenuti della lista possono essere ordinati per: - Inserimento; - Titolo; - Data di creazione; - Data di modifica. In abbinamento a questo parametro bisogna impostare anche la direzione (Ascendente, Discendente)

### Scrolling a destra e a sinistra

Un'altra funzionalità importante di questo widget è quella di visualizzare il link al contenuto precedente e successivo rispetto allo slug passato nella URL. La comunicazione del widget dovrà essere attiva e si dovrà impostare il selettore Scrolling al valore desiderato.

### Abilitazione dei Feed

Attraverso l'uso del widget `contentList`, i contenuti pubblicati sul sito possono essere resi disponibili anche in formato RSS (Really Simple Syndication). Dal pannello di configurazione sarà necessario abilitare tale funzionalità e dopo aver inserito il Titolo del feed, il numero di elementi da visualizzare e scelto il formato da utilizzare fra Atom 1.0, Rss 2.0, il widget riporterà, in basso a sinistra, una icona che consentirà l'estrapolazione del feed in formato XML.

Il link può essere copiato e incollato in un aggregatore di feed (Reader). I feed consentono di notificare la pubblicazione di un nuovo contenuto su Web e permette di aggregare facilmente i contenuti di diversi siti e presentarli sotto altre forme.

I feed rss elaborati rispetteranno fedelmente le impostazioni dei widgets, ed in particolare l'applicazione dei filtri per la selezione dei contenuti.

Nella stessa pagina più widgets contentList possono avere i feed abilitati.

---

**Importante:** Attenzione: i feed, se abilitati, sono fruibili anche nel famoso formato JSON Feed Version 1.0.

---

### Site Map

Il sistema consente la creazione automatica del file sitemap.xml secondo le specifiche del protocollo sitemap - ver. 0.9 - Il file Raccoglie tutti gli URL delle pagine e dei contenuti dei widgets «contentList» che hanno il sitemap abilitato. L'url delle pagine è di default inserito nel sitemap. Per i widget «contentList» il sitemap va abilitato dal pannello di configurazione del widget.