Symbiosis Institute of Computer Studies and Research

Subject : Design of CMS

Topic : CMS Case Study Assignment

Name : Amal Sunil Pillai

Prn: 19030121009

Class: Div A

Semester : V

1. Synopsis:

My choice of CMS for the case study is the ‘Bolt CMS’. It’s a free, open-source CMS based on PHP that was released in 2012 and developed by Two Kings and the Bold community. I chose this CMS specifically as it’s widely used by freelancers all over the world and known for its simplicity, flexibility. It strives to be as simple and straightforward as possible, it’s quick to set up, easy to configure and uses elegant templates.

Bolt is created for:

- End users: who focus on producing and editing content.
- Front-end developers / designers who want to set up websites where the CMS is flexible and easy to implement.
- Developers who need a system that's easy to manage, cheap, flexible and versatile.

Why use Bolt CMS ?

- It’s lightweight, it isn’t blown with a lot of features that people could need, it has a “do it as extension” principle which leads to cleaner CMS core.
- Fast as it’s lightweight, have optimized database queries and in-built cache system which can be used for cache generated thumbnails for example.
- User-friendly platform that is reliable and lets users deliver projects faster.
- Written using object oriented PHP which leads to clean and easy to maintain code.

2. Implementation:

Requirements :

The system requirements for Bolt are modest, and it should run on any fairly modern web server.

- PHP 7.2.9 or higher
- Access to SQLite (which comes bundled with PHP), or MySQL/MariaDB.

The PHP installation has a few additional requirements. On most servers these are installed by default, and Bolt should work out-of-the-box.

- A minimum of 32MB of memory allocated to PHP
- The following common PHP extensions:
    - pdo
    - mysqlnd (to use MySQL as a database)
    - openssl
    - curl
    - gd
    - intl (optional but recommended)
    - json
    - mbstring (optional but recommended)
    - opcache (optional but recommended)
    - posix
    - xml
    - fileinfo
    - exif
    - zip

Note: The following PHP modules are known to conflict with Bolt and it's underlying Symfony components, and must be disabled:

- Zend Guard Loader
- ionCube

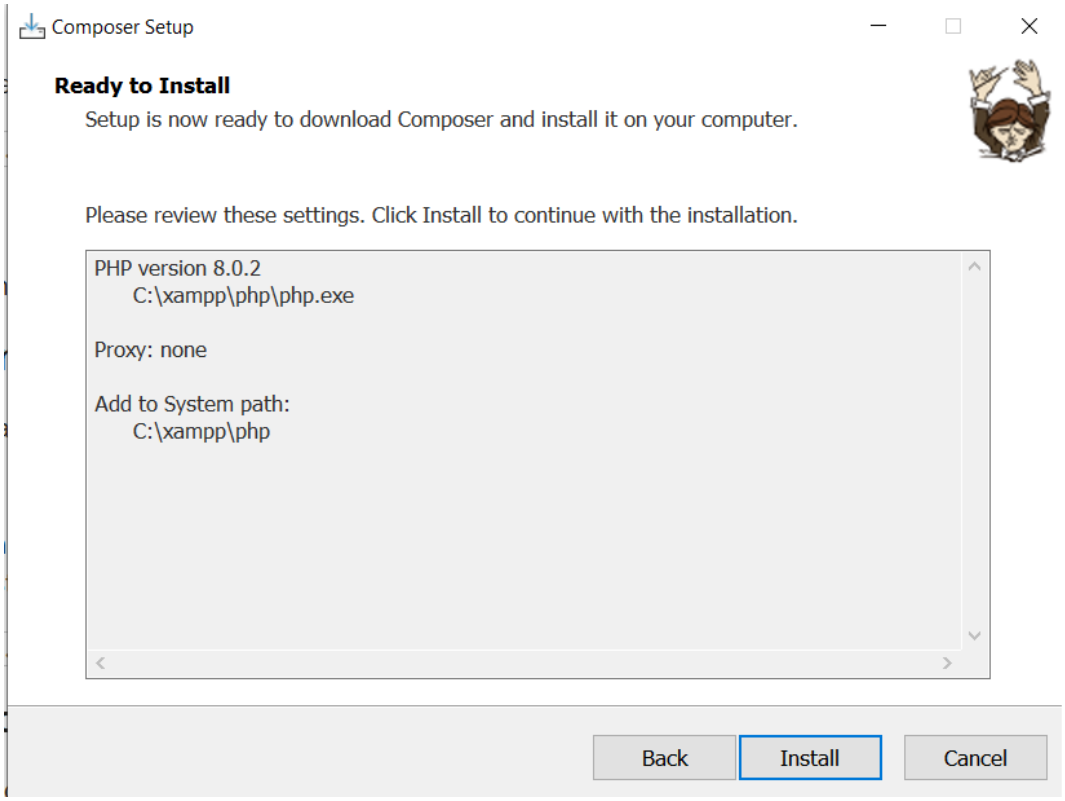
Browser requirements:

- Works with almost any modern web browser, both on Desktop and mobile. Internet Explorer is not supported, it’s recommended to use Chrome, Edge, Firefox or Safari.

Basic Installation process:

There's multiple ways to install Bolt, but it's recommended to do using 'composer create-project' as the fastest way to install the same and get it running.
Composer is a dependency manager for PHP, i.e. it helps us to manage the third-party libraries and tools(such as Bolt itself) that our project relies on.

Step 1: Installing composer
For composer to be installed we need PHP to be already installed in the system.



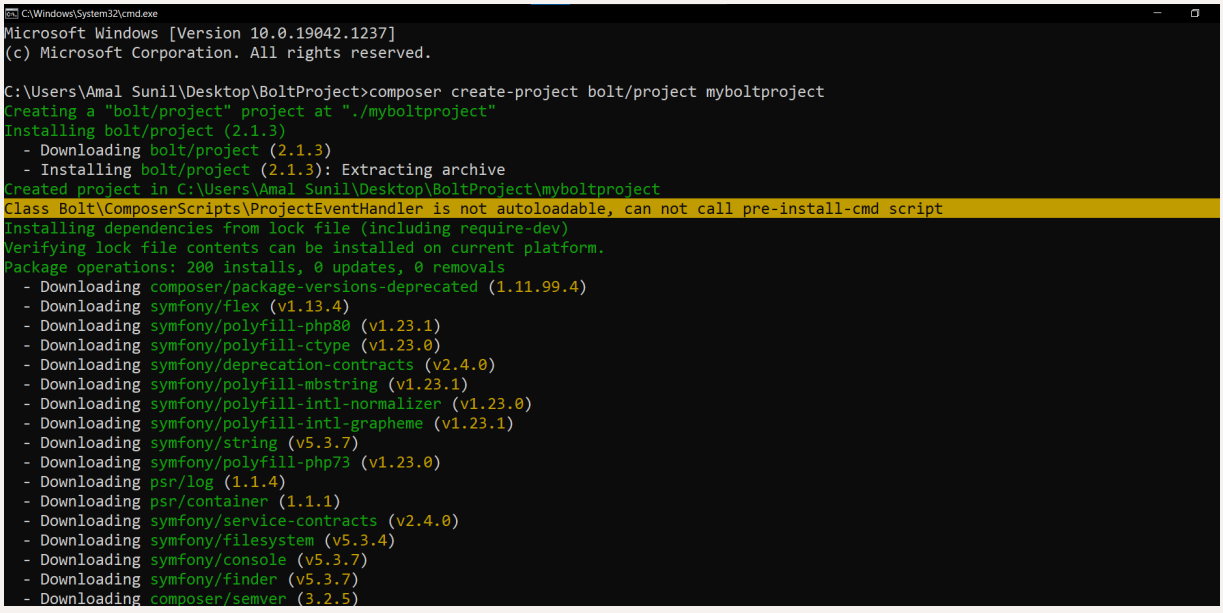
Using command line we can install the composer from 'getcomposer.org' or by using the following command in command line:

```
curl -sS https://getcomposer.org/installer | php
```

Step 2: Setting up new Bolt project

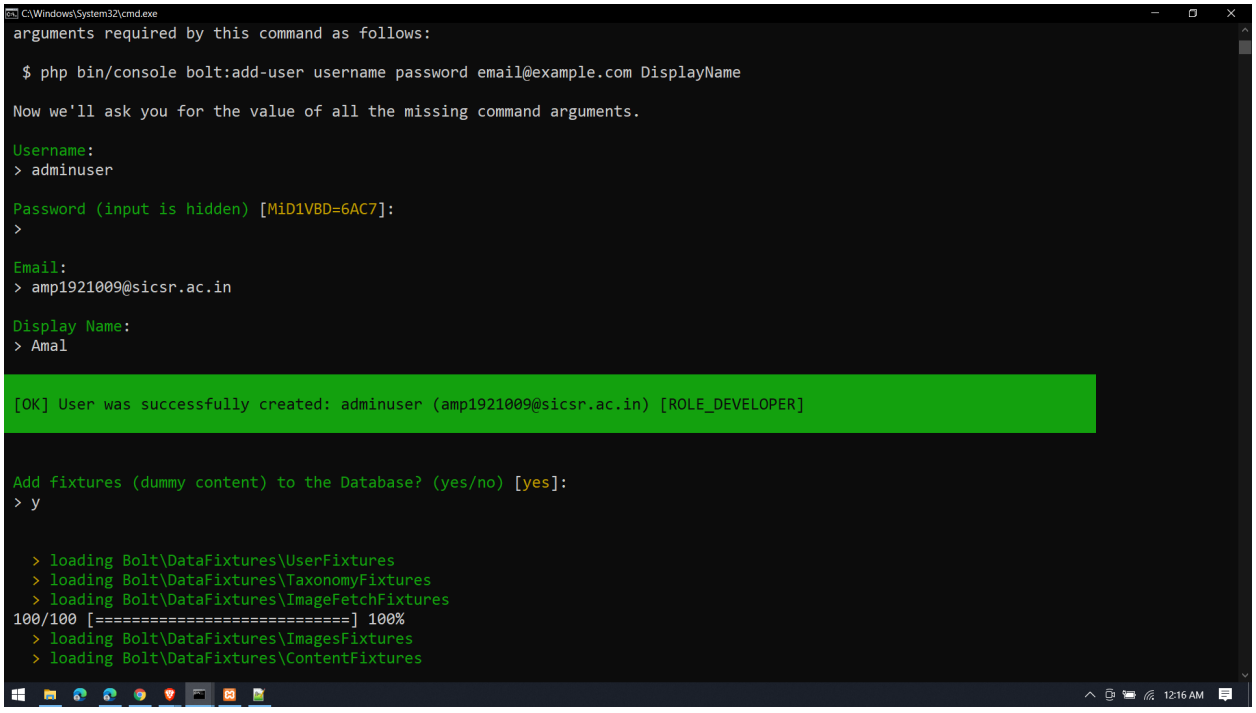
With the command line open in the folder inside of which we want to create our new Bolt project, we can use the following command :

```
composer create-project bolt/project [myprojectname]
```



Step 3: Configuring the database (Optional)

It’s optional to set up a database connection, Bolt by default will use SQLite database without any configuration needed.
I’ll be using MySql/MariaDB in my case, to do so we have to navigate into the newly created folder and configure the database in ‘.env’, replacing the db_user, db_password and db_name with the appropriate values that we have created.



Step 4: Initializing the new project

Option 1 :
Run `php ./bin/console bolt:setup` in the command line which will create and initialise the Database for us, then lets you create the first (admin) user, and add some dummy content ("fixtures") to the database.

Option 2:
Using separate steps

```
bin/console doctrine:database:create
bin/console doctrine:schema:create
bin/console bolt:add-user --admin
bin/console doctrine:fixtures:load
```

```
> ^C
C:\Users\Amal Sunil\Desktop\BoltProject\myboltproject>php ./bin/console bolt:setup
Created database `bolt_db` for connection named default

Add Bolt User Command
=====================

 If you prefer to not use this interactive wizard, provide the
 arguments required by this command as follows:

  $ php bin/console bolt:add-user username password email@example.com DisplayName

 Now we'll ask you for the value of all the missing command arguments.

 Username:
 > adminuser

 Password (input is hidden) [MiD1VBD=6AC7]:
 >

 Email:
 > amp1921009@sicsr.ac.in
```

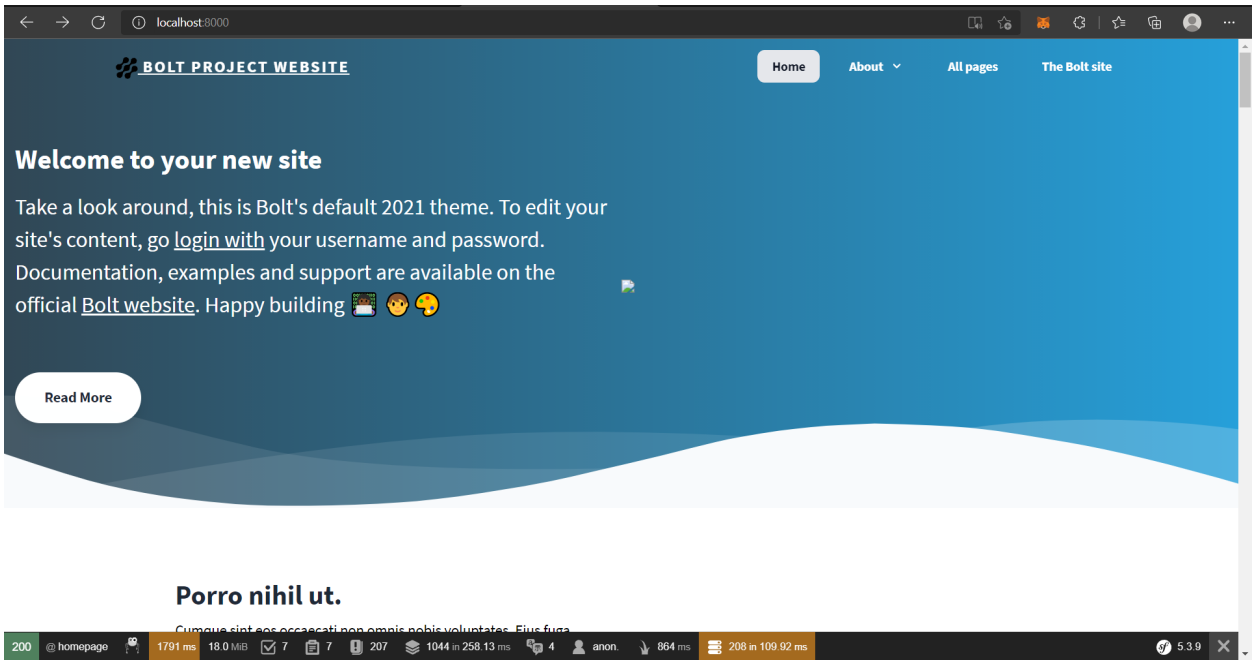
IMPORTANT: It's important that we create at least one admin user, otherwise we won't be able to log into the backend.

Step 5: Starting the server

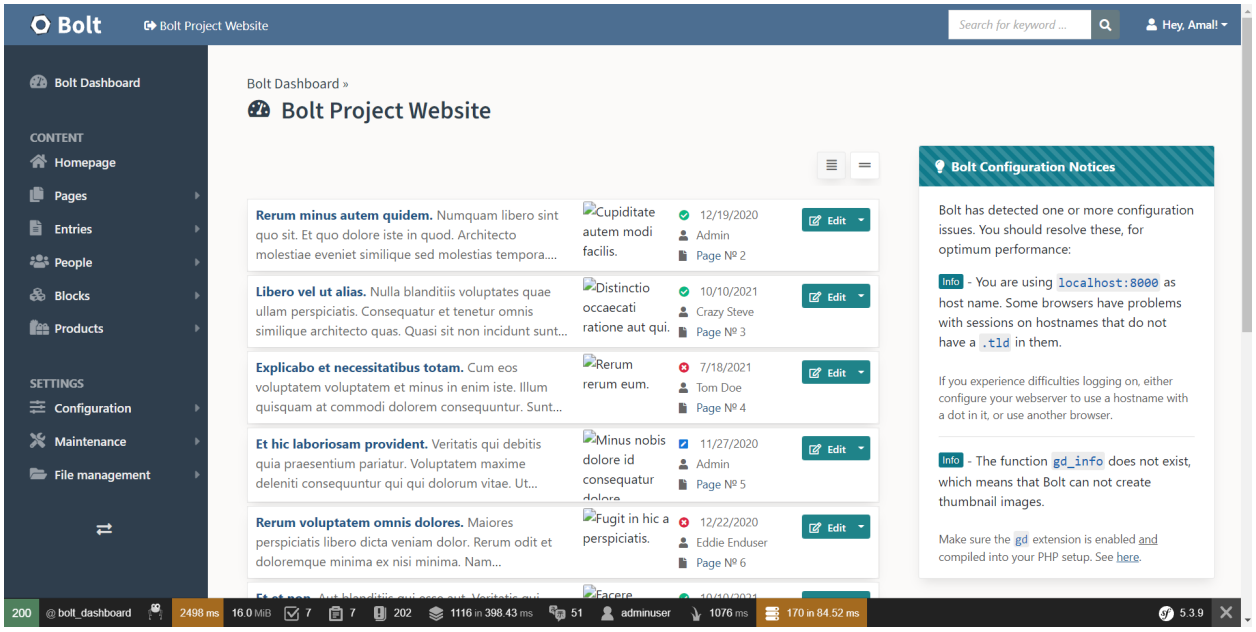
We can start PHP's built-in webserver by running:

```
php ./bin/console server:start
```

Now we'll be able to see the welcome screen of Bolt in our web browser in localhost with the configured port number.
We can log in using the credentials we created when setting up the 'first user.'

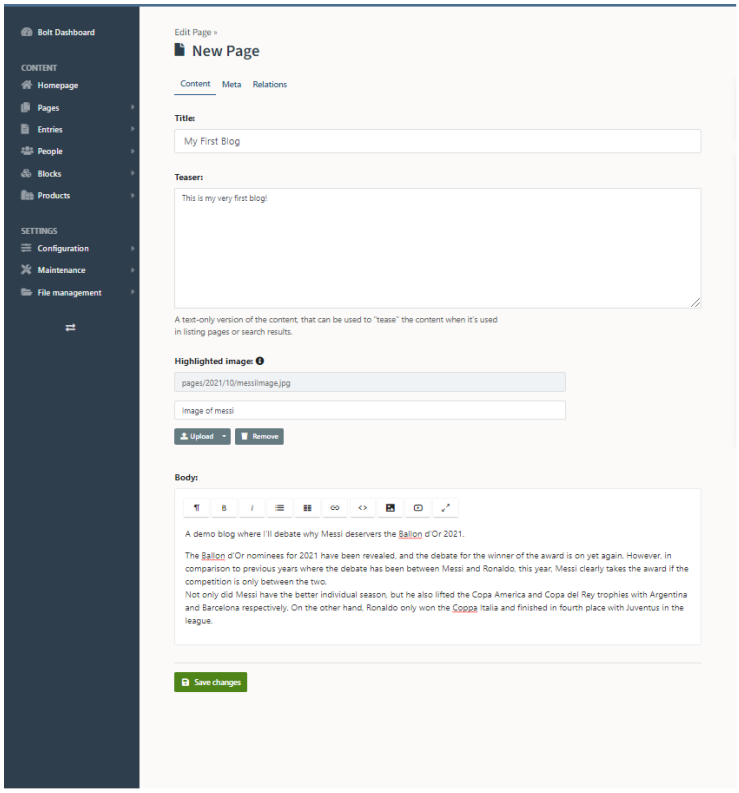


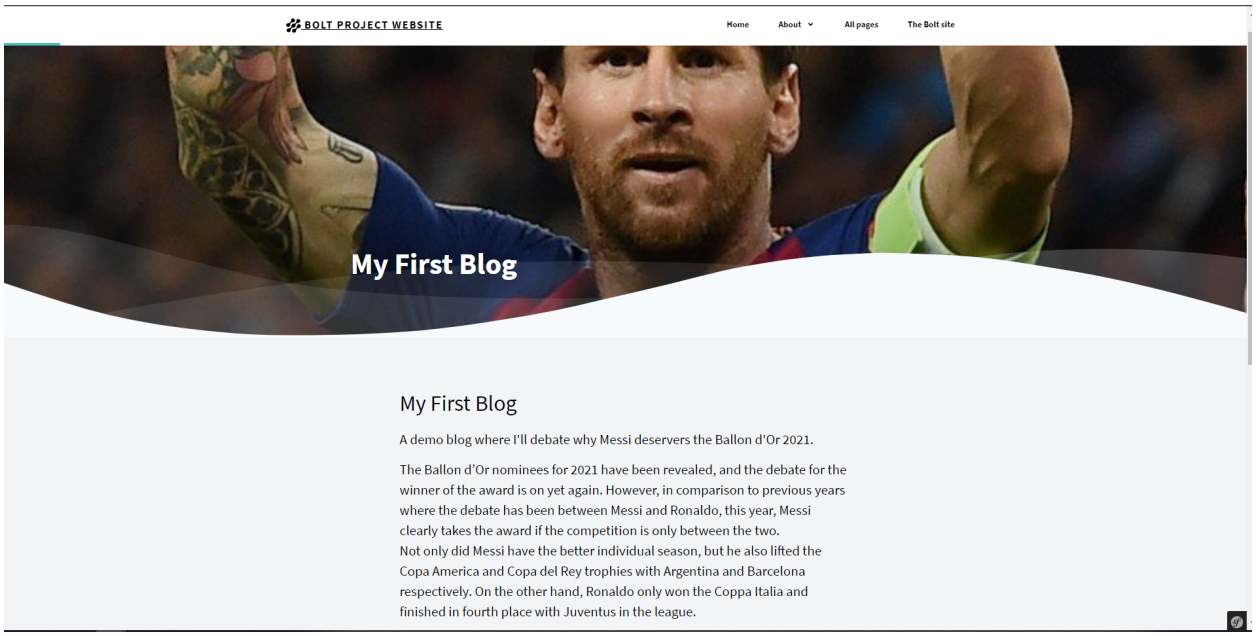
After login, Admin dashboard :



3. Usages :

Some basic usages and features:
Creating a sample page.





Features :

ContentTypes:

Bolt structures all its content in ContentTypes, a content type defines the structure of our website content, for example News, Blog Posts, Pages and sections are a ContentType.

Each ContentType we create gets its own database table with all the columns we define, this leads to a much cleaner and less blown database.
What makes them even better is the simplicity to define one, all content types are to be defined in a YAML format in the contenttypes.yaml file.

Example:

C:\Users\Amal Sunil\Desktop\BoltProject\myboltproject\config\bolt\contenttypes.yaml - Notepad++

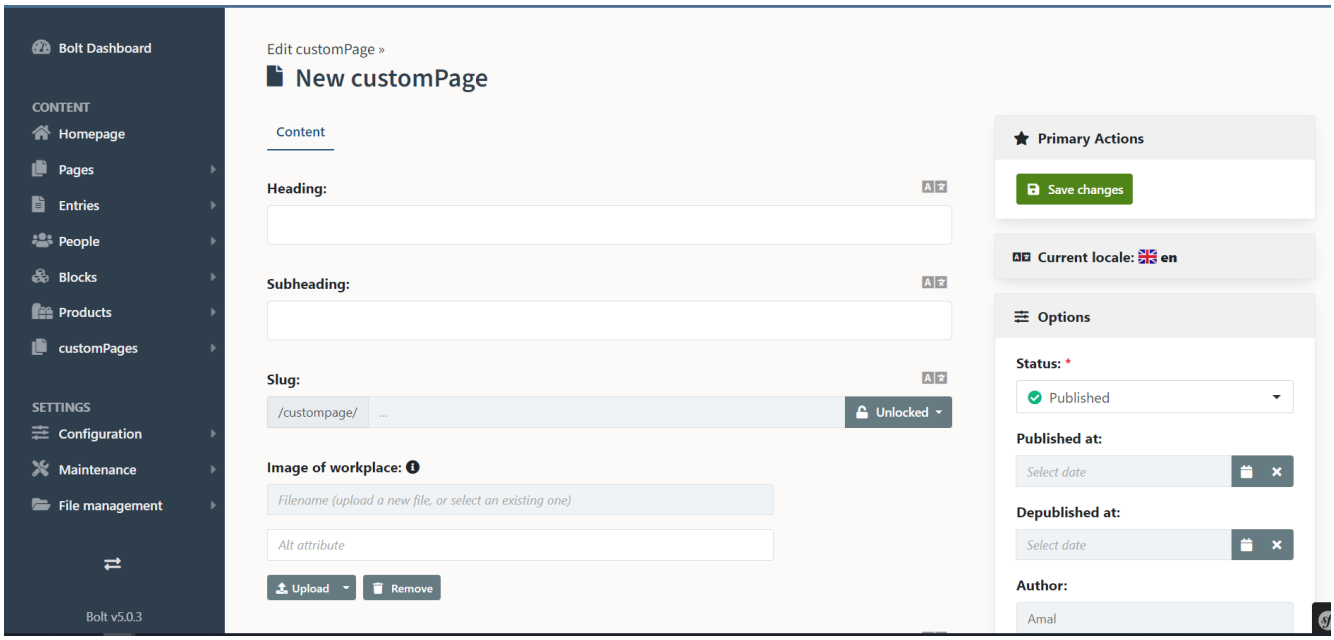
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

index.php | class-wp-rest-application-passwords-controller.php | README.md | style.css | main.dart | .env | console | post-create-project.php | contenttypes.yaml

```yaml
customPages:
    name: customPages
    singular_name: customPage
    title_format: [ heading, subheading ]
    fields:
        heading:
            type: text
            class: large
            group: content
            localize: true
        subheading:
            type: text
            class: large
            localize: true
        slug:
            type: slug
            uses: [ heading ]
            localize: true
        photo:
            type: image
            label: "Image of workplace"
        content:
            type: html
            allow_twig: true
            localize: true
    locales: ['en', 'nl', 'ja', 'nb']

# Possible field types:
#
# text - varchar(256) - input type text.
# number - double - Input field for numbers, with `mode: integer` or `mode: float`
# imagelist - text (65kb) - Input type for imagelists. Add multiple images. Useful for image sliders, galleries, etcetera.
# image - varchar(256) - image select/upload widget, stored as filename.
```

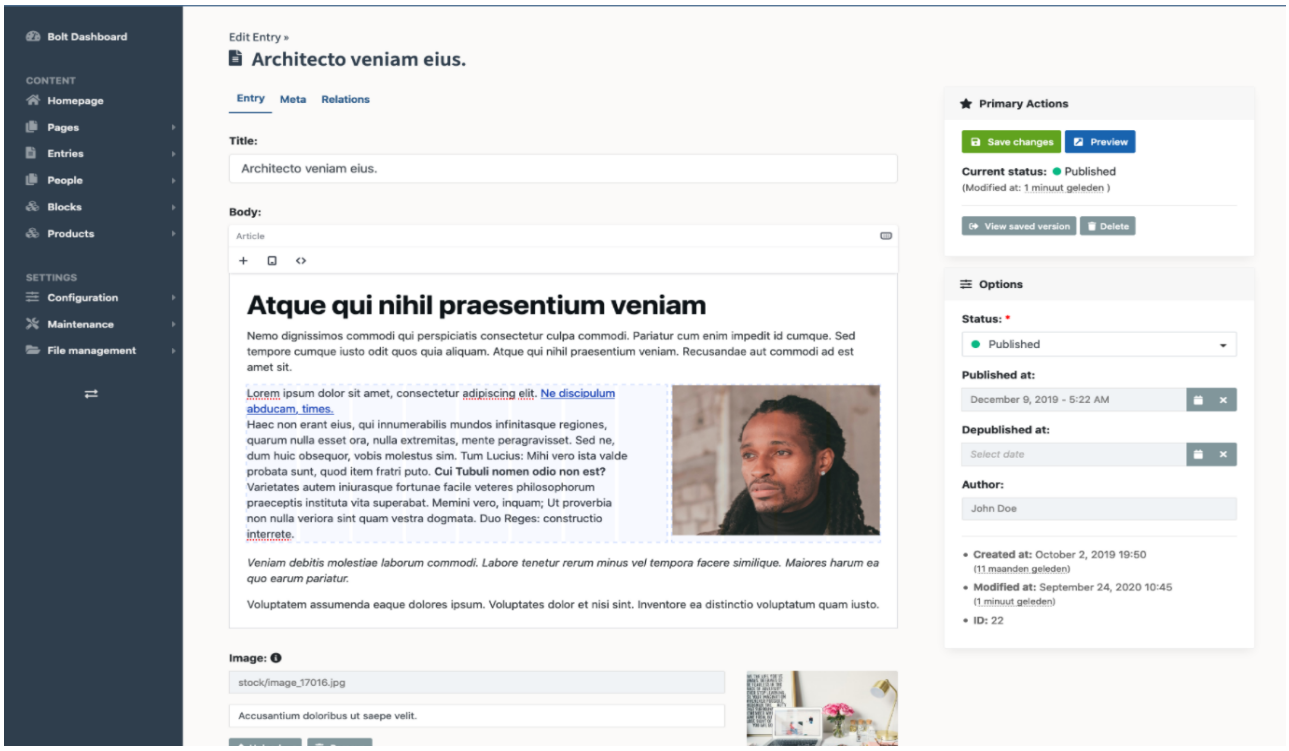
Now we can create a page using the above defined contentType



Bolt also has built-in internationalisation i.e. we can define in the content type if the pages has to be made translatable.

Dummy content :
We can easily add dummy data for test purposes or to showcase our work to a client to show how the end result will look.

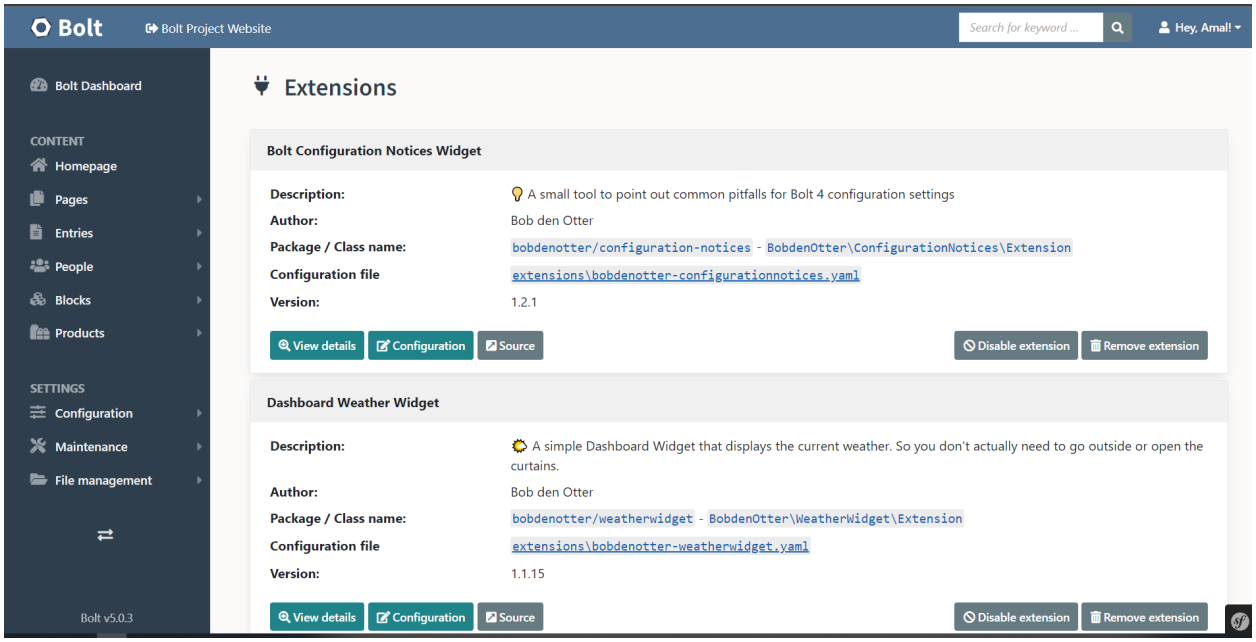
Feature rich content editors:
Bolt provides some of the best content editors: Article and Redactor, which are fully licensed for use in Bolt itself for free.This makes creating and adapting content easy.



Twig templates :
Twig is a flexible, fast and secure templating language that compiles to optimized PHP code. It makes it easy to generate HTML for our pages.

Rest and GraphQL:
Without the requirement of any additional configuration, Bolt provides a RESTful and GraphQL API, allowing us to use CMS in a headless mode(i.e. API only) where the CMS data can be used by any platform that has the permission and wants to listen to it.

Extensions:
Using Composer it’s easy to add functionality to Bolt through extensions.



Bolt also has a really good documentation and are going through frequent updates and are trying to reinvent the wheel for creating efficient and simple to use CMS.

4. Findings :

- As per my research and experience, Bolt provides one of the most user-friendly interface to build and manage CMS. They have a straightforward design and configuration methods to achieve instant results.

- Bolt embodies the slogan “Simple, sophisticated and straightforward”, meaning that they use very sophisticated technology to achieve a system that’s simple, straightforward and evident to the consumer side.

- Bolt is best suited for medium sized websites.

- Bolt uses Travis CI, for automatically running our unit tests and checking against various PHP versions, as they’re rapidly coming up with updates this helps them out for continuous integration and continuous inspection. The configuration is located in the .travis.yml file. The code standard, as well as some other helpful tools to get metrics about the codebase are run by Scrutinizer CI. The configuration is located in .scrutinizer.yml.

- Compared with Wordpress which is not a very modern PHP CMS and does not use Symphony or other components, Bolt CMS is a great alternative for developers looking for a modern PHP system that is less complex to replace Wordpress for example. Bolt is more straightforward and beginner friendly to create some really performant apps.

- A major disadvantage of using Blot is it's limited community size, and rapid changes in versions from the last few years which made it at times confusing for myself at times about certain features.

- Another disadvantage would be developer availability compared to major CMSs like WordPress, Drupal, Joomla etc.

5. Discussions :

Some of the issues that I faced and it's possible solutions :

1. Installation could be made even simpler :
    - The installation process was not easy as just downloading a installer package for a specific OS, as Blot depends on Composer dependency manager and other packages, users will have to install each component one by one as installation of one component assumes the presence of another component(as they're dependent).

    - The primary or recommended method for installation is using the command line commands, which I believe is not very beginner friendly.

    - Possible Solution : To make a installer package of zip file that consists of all the dependencies, so that beginners who are not used to working with CLI will find it easy, simplifying developer experience and attracting beginners will be a great way to get recognition in CMS platforms where it's the leaders like Wordpress that power 1/3rd of the websites and attracts more users due to its large community.

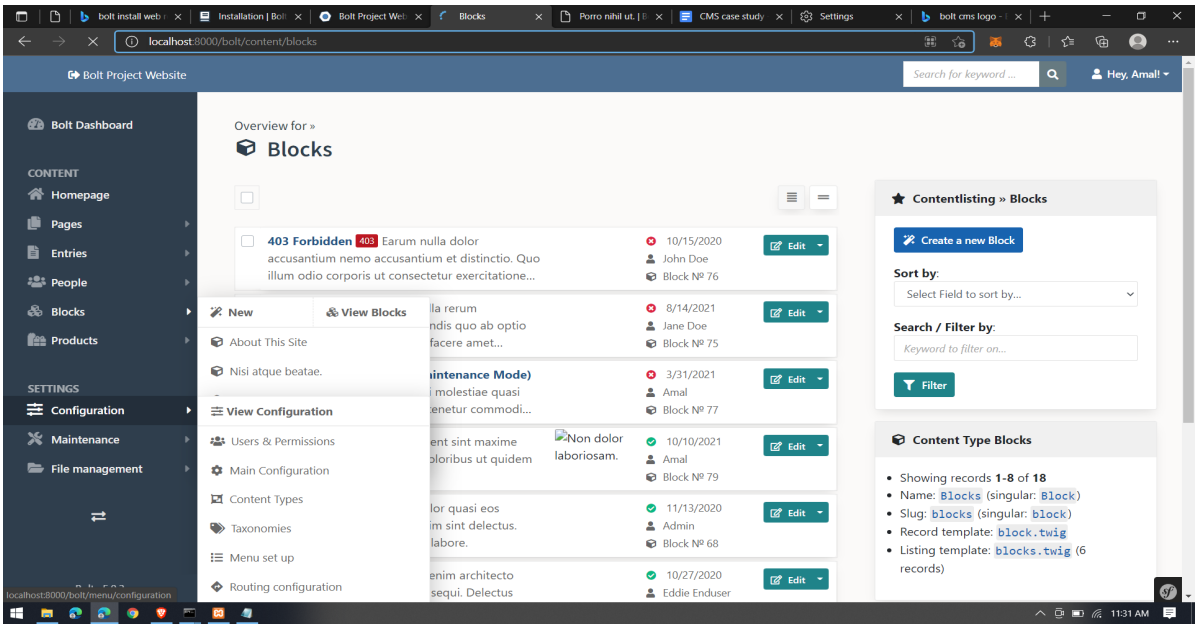
2. Limited community :
    - Due to a limited community I found it difficult to find answers for some of my queries.

- Even though they have good documentation, the lack of developers with knowledge in Blot CMS makes it difficult for beginners or students to find tutorials, support etc.

- Possible Solution : I think it's just about time when modern PHP CMS that is less complex than Wordpress, Drupal will attract users. I myself would prefer to work with Bolt due to its simplicity, if the latest version could have more resources, tutorials etc, I believe it would attract more developers to check out this CMS.

3. Rapid core changes in the latest versions :
    - Due to the rapid development changes in its core Blot gave me a tough time finding resources and reviews for the latest version.

4. A buggy dashboard UI at times :



    - The UI of the dashboard at times was buggy, some elements were overlapping with the drop down values of a previously selected item. It was not much of a problem given the simplistic design of the dashboard.

5. Slow loading of dashboard and website :
    - One of the major issues I faced was slowness in loading the dashboard and the website, this happened quite frequently even

after clearing the cache files. This made working with the platform feel laggy at times.

- Possible Solution : This would be something that would make the development process tiring, mainly when other CMSs that I have worked with too had similar issues, Blot appears to be blazing fast or straight the opposite at times. I believe the solution to this would be checking the integration of the core components that were brought in the latest version(Blot 4), because this was not a problem that I experienced in the Blot 3.7 version. Otherwise it would be a problem only some users face, in that case having a large community for support would be able to handle these issues.

6. Conclusion :

Bolt being a modern PHP CMS rich features like contentType structures, built-in internationalisation, Out-of-the-box API, easy to add dummy content, feature rich content editors etc. and also giving it a simplistic and straightforward design to work with is what an Ideal CMS sounds like. Bolt CMS is a great alternative if not the best for developers looking for a modern PHP system that is less complex to replace Wordpress for example. Bolt has one of the most simple, straightforward and beginner friendly yet rich in features and powerful.

The major issue I believe with other CMSs like Blot itself is its limited community, more developers with knowledge about the platform is needed to make it big.

7. Recommendations :

As from the conclusion I believe that Blot will make it big in the coming years with a larger community support, I have read some of the reviews from developers who have migrated from top CMSs like Drupal, Wordpress have started to work with Blot that being one of the reason for me to choose Blot as my case study topic.Blot has an intuitive and easy to understand design, making it perfect for beginners to start with.

With the upcoming updates and a larger community support, Blot certainly has a big role to play in the future of CMS.

8. References:

https://boltcms.io/
https://docs.boltcms.io/5.0/getting-started/introduction
https://medium.com/bolt-cms/bolt-4-the-awesome-cms-225bbb9526b9
https://github.com/bolt/bolt