

1 WordPress

1.1 Why WordPress?

I chose WordPress because I like the name. I selected one that is well known and has a large community behind it. WordPress is also one of the most popular CMS (Content Management System) in the world, it is open source so it is free to use. WordPress is also very easy to use making it user friendly for both beginner, advanced developers and individuals of small companies.  Because WordPress is so popular, there are already many plugins written that you can easily use in your project completely for free.

WordPress also has an impressive market share it powers more than 43.5% of all websites on the internet!  This means there is always support available if you have questions or any problems with installing or errors.

1.2 The Main Concepts of WordPress

The Functions

WordPress offers the ability to customize page layouts from themes or custom template files. This allows you to bypass the default design of the website. There are many plugins available that provide functions like spam filters or login pages, this makes the usage much easier without knowing how to make those plugins yourself.knowledge.

WordPress also has a user management system that is built into it. This allows you to have multiple authors and administrators with different access levels to each of them. This usage is perfect for larger websites with multiple contributors. 

The Themes

WordPress also supports a lot of themes that can be easily installed and switched from via the Dashboard or the FTP. Themes can change the appearance and functionality of the site. You can get theme's for free, or you can pay for theme's(same with the plugins).

It is super handy to know that there are thousands of free themes available, but there are also premium themes that often offer more features and better support. The positive side of a theme is that a good theme can make your website look professional without having to have troubles with the code. 

The Plugins

There are enormous amount of plugins available in the WordPress.org database. Just like the Themes, there are free and paid themes on WordPress. Some of the most popular plugins of WordPress are:

- [Elementor Website Builder](#)
- [Automatic YouTube Gallery](#)
- [Jetpack – WP Security, Backup, Speed, & Growth](#)
- [Akismet Anti-spam: Spam Protection](#)

I also searched a bit in the plugin directory, and I found a plugin that in my opinion is super awesome:

- [the Dinosaur Game](#)
-

The Widgets

Widgets in WordPress are small functionalities that can be added to the sidebar and footer of a website via drag and drop system. These widgets can contain text, images, links, and other content. There are a lot of useful usages of widgets

- A search bar
- Recent posts
- Recent comments
- Categories
- Archives
-

The handy thing about widgets is that you can add them to your website without writing a single line of code. So it is a simple way to customize your website so it suits your needs. 

The Gutenberg Editor

In the newer version of WordPress you can use the Gutenberg editor. It is a block based editor that makes creating pages and posts much easier. It is a lot like the drag and drop system of the widgets. It gives you much more control over the layout of your website/content. 

1.3 Popular Websites Using WordPress

- Microsoft 
- NASA 
- Harvard University 
- The Walt Disney Company 
- Sony Music 
- The New York Times 
-

WordPress is flexible enough to power everything from small blogs to super large websites. The adaptability of WordPress makes it an awesome choice for almost any project! 

2 Using WordPress with Docker

I chose to create a WordPress website with MySQL and PhpMyAdmin so you can see that I know what I'm doing.

Docker makes it super easy to run WordPress and all the necessary services in containers therefore making installations and management much easier. 

2.1 Install Docker Compose

To use WordPress on Docker we need to first install Docker itself. If you have followed the previous Docker documentation, you already have it installed. If you did not, then go to this link and download it.

[Docker Desktop Installation](#)

Docker Desktop is available for Windows, macOS, and Linux, so you can install it on any operating system you use. The installation process is super simple and super good documented. 

2.2 Installing and Reviewing Dependencies

We need to use 3 official images for this tutorial, the images I will use are:

1. WordPress 
2. MySQL 
3. PhpMyAdmin 

We use PhpMyAdmin to be able to view the tables we have in the database.

These three services work together to create a fully functional WordPress installation:

- WordPress contains the CMS software itself
- MySQL stores all the data on a database
- PhpMyAdmin provides a graphical interface to manage the data on the database

2.3 Creating the yml file

If you have installed Docker Desktop then you need to start it before we continue. We need to do that, because otherwise the commands that we will use, they will not be recognised. After you have done that, you need to create a new yml file called:

```
docker-compose.yml
```

Then you need to add the following lines in the yml file:

```
version: '3'

services:
  # Database
  db:
    image: mysql:5.7
    volumes:
      - db_data:/var/lib/mysql
    restart: always
    environment:
```

```
MYSQL_ROOT_PASSWORD: password
MYSQL_DATABASE: wordpress
MYSQL_USER: wordpress
MYSQL_PASSWORD: wordpress
networks:
- wpsite
# PhpMyAdmin
phpmyadmin:
depends_on:
- db
image: phpmyadmin/phpmyadmin
restart: always
ports:
- '8080:80'
environment:
PMA_HOST: db
MYSQL_ROOT_PASSWORD: password
networks:
- wpsite
# WordPress
wordpress:
depends_on:
- db
image: wordpress:latest
ports:
- '8000:80'
restart: always
volumes: ['./:/var/www/html']
environment:
WORDPRESS_DB_HOST: db:3306
WORDPRESS_DB_USER: wordpress
WORDPRESS_DB_PASSWORD: wordpress
networks:
- wpsite
networks:
wpsite:
volumes:
db_data:
```

This code installs WordPress, MySQL, and PhpMyAdmin so we can use it in the Docker container. Once you have done that, you need to go to the directory where you created that file (in the terminal). And then you need to run this command:

```
docker-compose up -d
```

It will then install the dependencies from the yml file and then run them. The `-d` flag is to ensure the containers run in the background. ↗ If the command works fine you will eventually see this in your terminal. That means that everything is downloading.

The screenshot shows a terminal window within the VS Code interface. The title bar says 'Test' and 'WordPress.md'. The terminal tab is selected. The output shows:

```
50 ## 2.5 Making the yaml file
57
58 So if you have docker desktop installed, you need to run it before we continue.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS

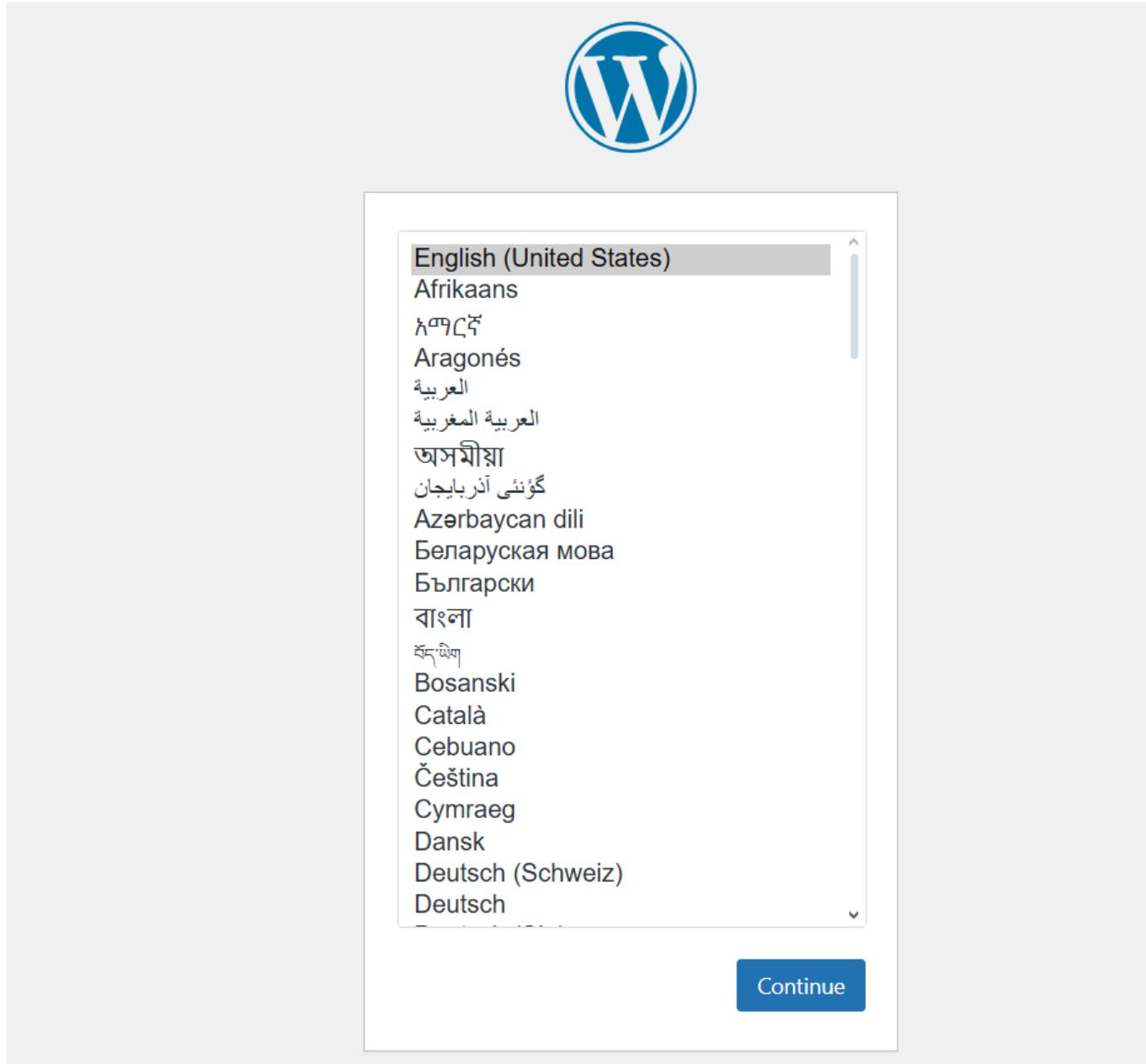
[+] Running 52/55
- wordpress [██████████] 244.6MB / 246.8MB Pulling
- db [██████████] 135.6MB / 136.9MB Pulling
- phpmyadmin [██████████] 194.1MB / 196MB Pulling
```

2.4 The URL in the Web Browser 🌐

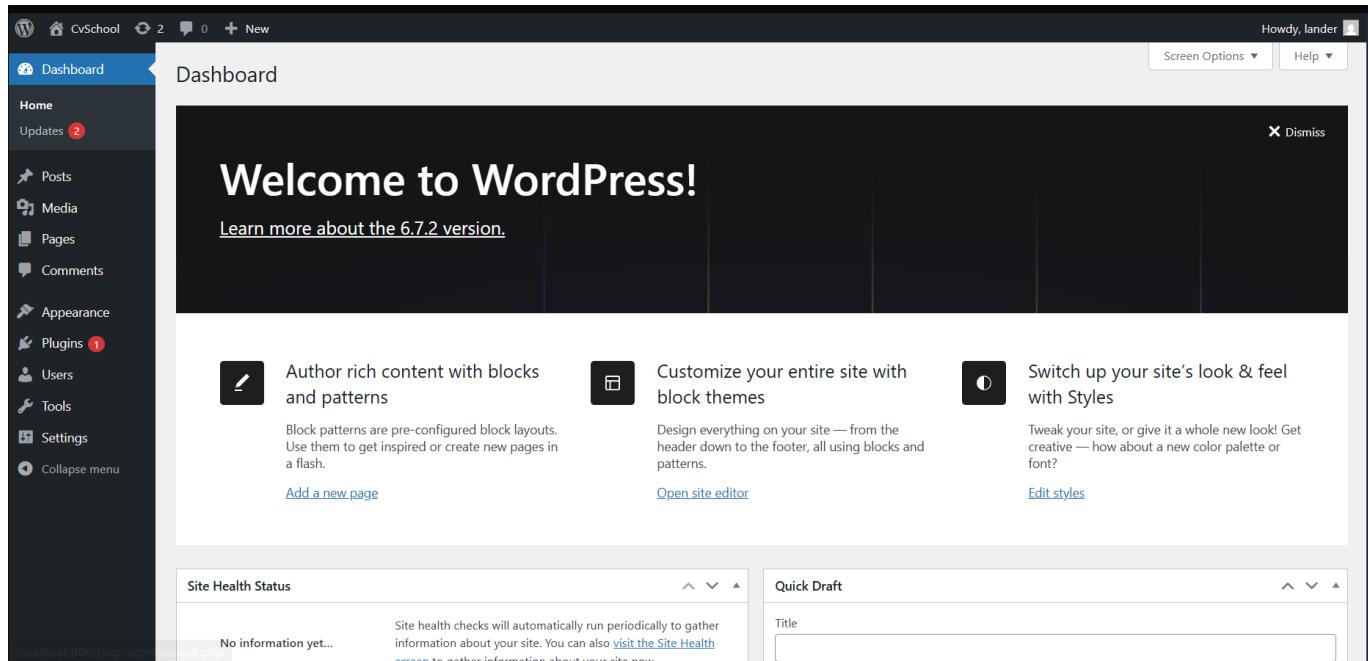
If you have done all of the steps above and the Docker container is running, then you need to go to this url where the docker is running:

```
http://localhost:8000
```

Then a window like this wil open:



In that view you can choose the language you want to use in WordPress. After that you need to follow the instructions on the screen to install WordPress. If that is done you need to log in with the username and password you set. If all of that worked correctly you will see a page like this appear:



Now you can go to the pages tab on the left, then you click on **add page**. Now you can begin your journey for creating awesome websites! 🎉

2.5 How To Access PhpMyAdmin 🔎

If your WordPress server is correctly running, and you want to see and manage the database. You need to go to the url where PhpMyAdmin is running at. It will be this url if you copied the yml file from above, otherwise you need to check the yml file for the port.:

`http://localhost:8080`

There you will need to log in with your username and password. Then you can edit, view and create tables in the database. Good luck! 🎉

2.6 Customizing and Developing WordPress 🎓

Now that you have WordPress installed with Docker, you can start customizing your website. You can do this in some ways, but the most popular are these:

1. Installing and customizing themes 🎨
2. Adding plugins for extra functionality 🏃
3. Creating pages and posts with the Gutenberg editor ✍️
4. Adding widgets to your sidebar and footer 📦
5. Creating menus for navigation 🌐

You can also develop your own themes and plugins if you want more control over the functionality and the layout of your website. WordPress also has extensive documentation for developers that you can use to guide you through the process. 📕

<https://developer.wordpress.org>

2.7 The Benefits of WordPress with Docker

Using WordPress with Docker has a lot of advantages:

1. **Isolation:** Each service runs in its own container, so they do not affect each other. 
2. **Portability:** You can easily move your entire development environment to another computer or device. 
3. **Version control:** You can use specific versions of WordPress, MySQL, and PhpMyAdmin in your containers. 
4. **Scalability:** It is easy to scale for larger projects. 
5. **Quick setup:** With one command you can set up a fully functioning WordPress environment/website. 

2.8 Troubleshooting

If you encounter any problem with your WordPress Docker installation, here are some common issues and solutions:

Cannot connect to the database 

You need to check if the MySQL container is running with. You can do this with this command:

```
docker ps
```

If the container is not running, you can try this:

```
docker-compose up -d db
```

Permission problems with files 

If you have issues with the file permissions, you can try this command:

```
chmod -R 777 ./
```

Note: This command is not recommended for production environments due to security risks.

Stopping Docker containers 

When you are done working, you can stop the containers with this command:

```
docker-compose down
```

And if you want to start these docker containers again later, you can do that with this command:

```
docker-compose up -d
```

3 Making My Own CV Website in WordPress

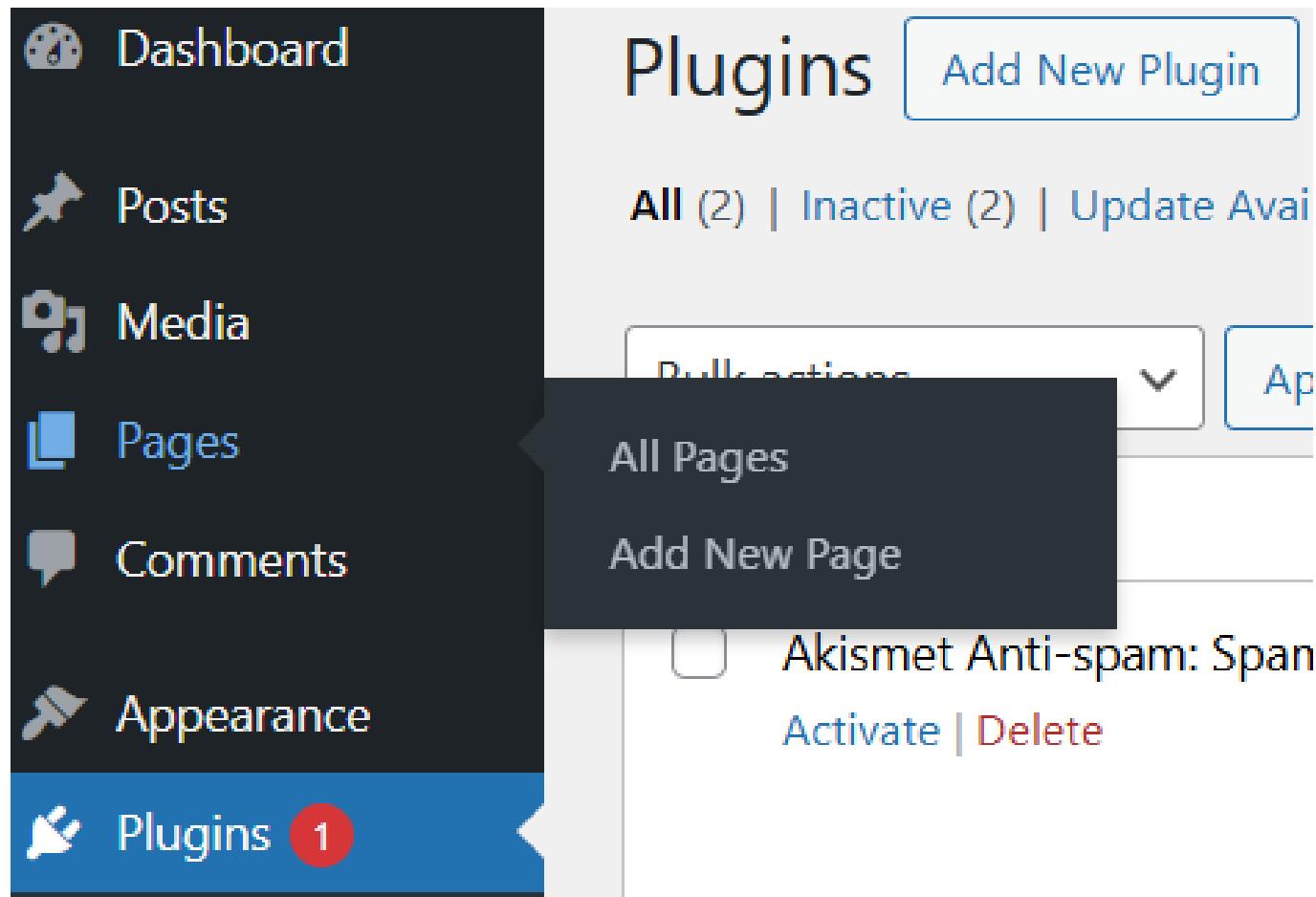
3.1 Setting Up Your CV Website

You first need to complete all the steps in the previous sections to install WordPress in Docker. Once you have WordPress up and running, we can proceed with creating a professional CV website.

3.2 Creating Your First Page

To start building your CV website, you need to add a new page. You do this by:

1. Navigating to the **Pages** tab on the left sidebar of your WordPress dashboard
2. Clicking on the **Add New** button at the top



After clicking it, you will see an empty page editor like this:

The screenshot shows the WordPress editor interface. At the top, there's a toolbar with icons for back, forward, and search. The title bar says 'No title · Page'. On the right, there are buttons for 'Save draft', 'Publish', and other settings. Below the toolbar, the main area has a placeholder 'Add title' and a note 'Type / to choose a block'. A large blue '+' icon is in the center. To the right is a sidebar with a table of page metadata: Status (Draft), Publish (Immediately), Link (/14), Author (lander), Template (Pages), Discussion (Closed), and Parent (None). At the bottom, the browser address bar shows 'localhost:8000/wp-admin/edit.php?post_type=page' and the system tray indicates the date and time as 3/18/2025.

3.3 Uploading Your CV as a PDF 📄

Now we need to give your page a title and upload your CV, we do that by:

1. Entering a descriptive title such as "My CV" or "Professional Resume"
2. Clicking on the blue + icon in the top left of the editor
3. Searching for and selecting the **File** block option

The screenshot shows the WordPress editor with a title 'CV Website Lander Cuypers'. The sidebar on the left has tabs for 'Blocks', 'Patterns', and 'Media'. Under 'Blocks', there are categories: 'Preformatted', 'Pullquote', 'Table', 'Verse', and 'Classic'. Under 'MEDIA', there are categories: 'Image', 'Gallery', 'Audio', 'Cover' (with a red circle around the 'File' icon), 'Media & Text', and 'Video'. The 'File' icon is highlighted with a red circle. The right sidebar shows the page metadata: Status (Draft), Publish (Immediately), Link (/cv-website-lander...), Author (lander), Template (Pages), Discussion (Closed), and Parent (None). A red box highlights the 'Move to trash' button at the bottom of the sidebar.

Once you have added the File block, you need to:

1. Click on the **Upload** button
2. Select your CV PDF file from your computer
3. And wait for the upload to complete

The screenshot shows the WordPress editor interface. On the left, there's a sidebar with categories: Blocks, Patterns, Media, DESIGN, and a preview area. The main content area has a title "CV Website Lander Cuypers" and a media block with a file upload interface. On the right, there are settings panels for Page, Block, and DESIGN.

Page: File
Add a link to a downloadable file.

Block:

- Color:
 - Background
 - Link
- Dimensions
- PADDING
- MARGIN
- Border

After it was a successful upload, you will see your CV displayed on the page like this:

The screenshot shows the published version of the CV website. The main content area displays the uploaded CV document. On the right, there are settings panels for Page, Block, and DESIGN.

Page: File
Add a link to a downloadable file.

Block:

- PDF settings
 - Show inline embed
- HEIGHT IN PIXELS
- Settings
 - LINK TO
- Advanced