

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

(повне найменування назва факультету)

Кафедра прикладної математики

(повна назва кафедри)

Магістерська робота

Проектування та розробка структури системи пошуку роботи

Виконав: студент групи ПМПм-22

спеціальності

113 Прикладна математика

(шифр і назва спеціальності)

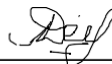


Горак О.Ю.

(підпис)

(прізвище та ініціали)

Керівник



Дяконюк Л.М.

(підпис)

(прізвище та ініціали)

Рецензент



Бернакевич І.Є.

(підпис)

(прізвище та ініціали)

Львів – 2022

Зміст

Вступ

Магістерська робота	1
1. Формулювання проблеми	5
1.1 Загальне формулювання проблеми	5
1.2 Функціонал Адміністратора	5
1.3 Функціонал Студента	5
1.4 Функціонал Роботодавця	6
1.5 Функціонал Рекрутера	6
1.6 Аналоги систем пошуку роботи	7
2. Розробка системи за допомогою Вордпресу	7
2.1 Актуальність технології вордпрес	7
2.2 Переваги та недоліки Вордпресу	8
3. Використання плагінів у проекті.....	9
3.1 Розширені спеціальні поля	9
3.2 Оксиджен.....	10
3.3 Менеджер роботи.....	11
3.4 Конструктор резюме	12
3.5 Багатомовність	13
3.6 WP Cerber Security	14
4. Двонаправлений зв'язок	14
4.1 Двонаправлений зв'язок загальні поняття. Фрагменти	14
4.2 Динамічне відображення полів	18
5. Редагування функціоналу плагінів в коді.....	19

5.1	Структура проекту. Сі-панел.....	19
5.2	Доповнення плагінів	21
6.	Розробка моделі даних для системи.....	24
6.1	Концептуальне проектування бази даних	24
6.2	Логічне проектування бази даних.....	24
6.3	Фізичне проектування бази даних	24
7.	Результати реалізації системи пошуку роботи.....	26
8.	Висновок.....	29
	Використані джерела	30

Вступ

У сучасному світі завжди приділяють увагу більш оптимальному розв'язуванню проблеми пошуку та аналізу інформації. Кожному користувачеві пошукової системи необхідно отримати результати, які відповідають його потребам. Таку проблему можна розв'язати шляхом аналізу та фільтруванню даних і видачі результатів. Загалом це можна розглядати, як проблему аналізу інформації.

Серед вищезгаданих проблем можна виділити таку підгрупу, метою розв'язування яких, є оптимізація пошуку роботи. Зокрема, у цій роботі розглядаються методи оптимізації системи пошуку роботи та створення зручного середовища для роботодавців. Студенти, які використовують існуючі системи пошуку роботи, витрачають багато часу на знаходження вакансії, яка б відповідала їх побажанням. Також є ймовірність натрапити на недобросовісного роботодавця. Роботодавці при пошуку людей часто натрапляють на

некомпетентні кадри, що певним чином сповільнює пошук кандидата на вакансію. Тому варто розглянути варіант розв'язування даних проблем більш оптимальним способом. Зокрема, можна зменшити час пошуку вакансії, чи кандидата, оптимізувавши пошук роботи та розміщення вакансій в системі.

Об'єктом дослідження цієї роботи є система накопичення та фільтрування інформації, що спрямована на спрощення комунікації студентів з роботодавцями. Предметом дослідження є аспекти реалізації цієї проблеми на практиці.

Мета: вивчення ключових елементів , які є основою системи пошуку роботи; проектування та розробка інформаційної системи підтримки сервісів кар'єрного розвитку університету та розширення потенціалу цієї системи.

Завдання магістерської роботи: формування основних параметрів проектування системи підтримки сервісів кар'єрного розвитку університету, опис взаємодії між елементами системи та поглиблення знань щодо роботи з системою управління контентом WordPress, використання інструменту Advanced Custom Fields для формування та оптимізації системи.

З метою поглиблення знань та практичної реалізації певної частини системи було створено початкову версію системи, розроблений загальний дизайн та взаємодію між елементами системи. У даній системі використовується база даних MySQL. Заповнення цієї бази даних відбувається через графічний інтерфейс. Працювати з базою даних можливо через графічний клієнт вордпресу. За допомогою інструменту Advanced Custom Fields було спроектовано загальну концепцію системи та створені певні сторінки для різних категорій користувачів.

Як основну особливість цієї роботи можна виділити призначення цієї системи, а саме – розробка системи пошуку роботи для Львівського національного університету.

У межах розгляду цієї проблеми можна виділити такі переваги реалізації системи на практиці :

1. Формування основи для системи пошуку роботи Львівського національного університету з подальшою оптимізацією цієї системи.

2. Спосіб реалізації основи цієї системи можна використати для врахування помилок у реалізації іншого функціоналу системи.

1. Формулювання проблеми

1.1 Загальне формулювання проблеми.

Необхідно розробити для Львівського Національного Університету ім. І.Франка закриту систему пошуку роботи для студентів та роботодавців. В даній системі необхідно надати доступ користувачам з наступними ролями : Адміністратор , Студент, Роботодавець та Рекрутер.

1.2 Функціонал Адміністратора.

Адміністрування сайту – функціонал який дає змогу забезпечити функціонування сайту, оновлення наповнення, розділення ролей, додавання нових користувачів та інших особливостей системи. Для цієї реалізації стоять такі вимоги для функціоналу адміністратора сайту:

- Зміна або присвоєння ролі користувача.
- Підтвердження запиту на реєстрацію.
- Наповнення розділу новин.
- Можливість редагувати розділи з партнерами.
- Адміністрування, наповнення та оновлення інших інформативних розділів(популярні запитання, створення нових елементів та редагування існуючих).

1.3 Функціонал Студента.

Дана категорія користувачів буде найбільшою, оскільки саме для студентів і створюється дана система пошуку роботи. Для цієї ролі потрібен наступний функціонал:

- Реєстрація у систему після заповнення форми, яку підтверджує

адміністратор сайту, або ж автоматичне підтвердження при використанні корпоративної пошти “user@lnu.edu.ua” .

- Створення резюме на сайті.
- Відгук на вакансію.
- Пошук вакансії за фільтрами.
- Додавання фото у портфоліо.
- Оновлення інформації профілю.
- Відновлення паролю.

1.4 Функціонал Роботодавця.

Однією з основних ролей є роботодавці , так як саме вони надають можливість працевлаштуватись. Окрім розміщення базових вакансій , є можливість додати стажування, навчальні програми , тощо. Для цієї ролі передбачений наступний функціонал:

- Реєстрація у системі після заповнення форми про компанію , яка підтверджується адміністратором.
- Додавання, закриття або архівування вакансії.
- Оновлення інформації про вакансію.
- Оновлення неактивної вакансії.
- Додавання, видалення рекрутера.
- Оновлення інформації про компанію.
- Відновлення паролю.
- Перегляд розміщених студентами резюме.

1.5 Функціонал Рекрутера.

Основним функціоналом рекрутера буде:

- Додавання , закриття або архівування вакансії.
- Оновлення інформації про вакансію.

- Оновлення неактивної вакансії.
- Відновлення паролю.
- Перегляд розміщених студентами резюме.

Головна	Про сайт	Для студентів	Розмістити резюме	Для роботодавців	Переглянути резюме	Актуальні вакансії	Програми стажування та практики	Наші партнери	Статті	Контакти
		Як скласти резюме	<i>Форма у яку студенти вносять інформацію про себе</i>	Умови співпраці	<i>Після реєстрації роботодавець може зайти і переглядати резюме студентів</i>	<i>Форма, яку заповняють роботодавці</i>		<i>Дода компаній із переходом на їх сайти</i>		
		Як написати мотиваційний лист		Інформація про компанії						
		Поради щодо пошуку роботи		Додати вакансію						

Рис1. Загальні вимоги до системи

1.6 Аналоги систем пошуку роботи .

Був проведений аналіз наступних систем пошуку роботи :

- <https://jobs.ua/>
- <https://studlava.com/>
- <https://rabota.ua/>
- <https://www.work.ua/>
- <https://grc.ua/>

Усі наведені вище системи пошуку роботи мають дещо схожий функціонал проте досить різний дизайн і зручність користування. Всі ці сайти містять типову структуру: сторінка пошуку роботи за критеріями, блок розміщення вакансій, сторінки вакансій з описом, сторінки додавання компанії. Різниця є не значною, але все ж додає індивідуальності сайтам. Тому при створенні власної системи пошуку роботи потрібно врахувати всі особливості системи та застосувати по можливості найкращі розв'язки відповідно до потреб.

2. Розробка системи за допомогою Вордпресу

2.1 Актуальність технології вордпрес.

Однією з найкращих безкоштовних платформ для створення веб-сайтів є вордпрес. В даній системі управління контентом використовується база даних MySQL та основний код написаний на PHP. Вордпрес надає багато корисних інструментів для базової розробки сайтів. З додаванням нових плагінів та тем дана система не втрачає актуальності. WordPress є досить універсальним та гнучким засобом обробки наповнення сайтів.

2.2 Переваги та недоліки Вордпресу.

Переваги :

- Однією з головних переваг вордпрес є те, що це безкоштовна система управління контентом. Всі файли та база даних сайту є у повній власності розробника. Хостинг можна обирати відповідно до необхідності сайту.
- Дуже хороша підтримка з боку розробників системи. Часті оновлення для покращення швидкості та надійності роботи
- Велика кількість плагінів для розв'язування різних завдань
- Зручна і проста адмін панель. Не зважаючи на власний досвід у використанні системи, досить просто інтуїтивно розібратись як додавати сторінки, розділи, новини.
- Простота установки та перенесення
- Можливість розширення сайту
- Висока швидкість роботи
- Безпека на дуже хорошому рівні

Недоліки :

- Є необхідність постійного контролю за сайтом, безпекою та коректністю роботи

- Якщо потрібно реалізувати специфічну архітектуру, то потрібно спершу з нуля все створити і вже тоді переносити у вордпрес.

- Велика кількість неякісних плагінів, які можуть спричиняти конфлікти з іншими. Тому це погіршує швидкість завантаження сайту та може понизити рейтинги сайту

- Майже всі безкоштовні плагіни містять тільки базові рішення, тому коли потрібні якісь особливості, потрібно купляти преміум версії плагінів, які будуть розширювати функціонал та покращувати гнучкість.

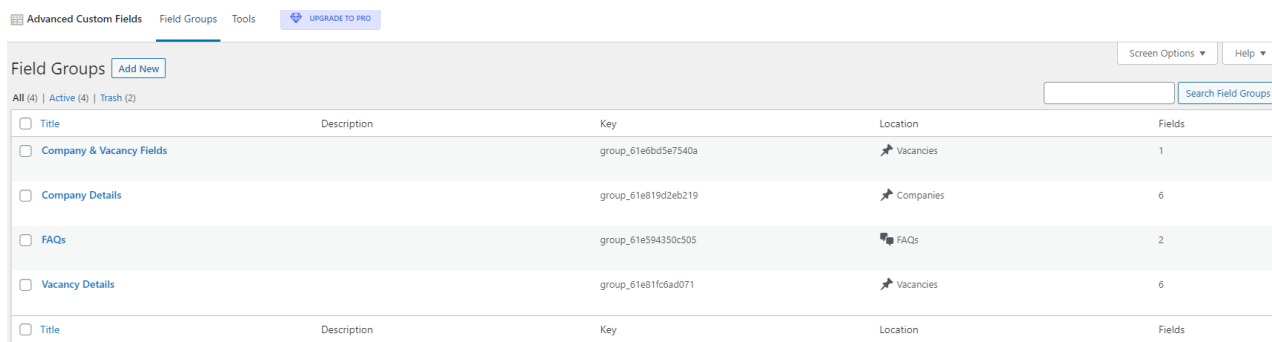
3. Використання плагінів у проекті

3.1 Розширені спеціальні поля

Основна сторінка плагіну: <https://www.advancedcustomfields.com/>

В даній системі одним з основних плагінів для налаштування полів є .

Даний плагін дозволяє додати додаткові поля які допоможуть краще заповнити сторінку всім необхідним. Також цей плагін використовується як допоміжний для різних двосторонніх зв'язків між групами елементів.



<input type="checkbox"/> Title	Description	Key	Location	Fields
<input type="checkbox"/> Company & Vacancy Fields		group_61e6bd5e7540a	Vacancies	1
<input type="checkbox"/> Company Details		group_61e819d2eb219	Companies	6
<input type="checkbox"/> FAQs		group_61e594350c505	FAQs	2
<input type="checkbox"/> Vacancy Details		group_61e81fc0ad071	Vacancies	6
<input type="checkbox"/> Title	Description	Key	Location	Fields

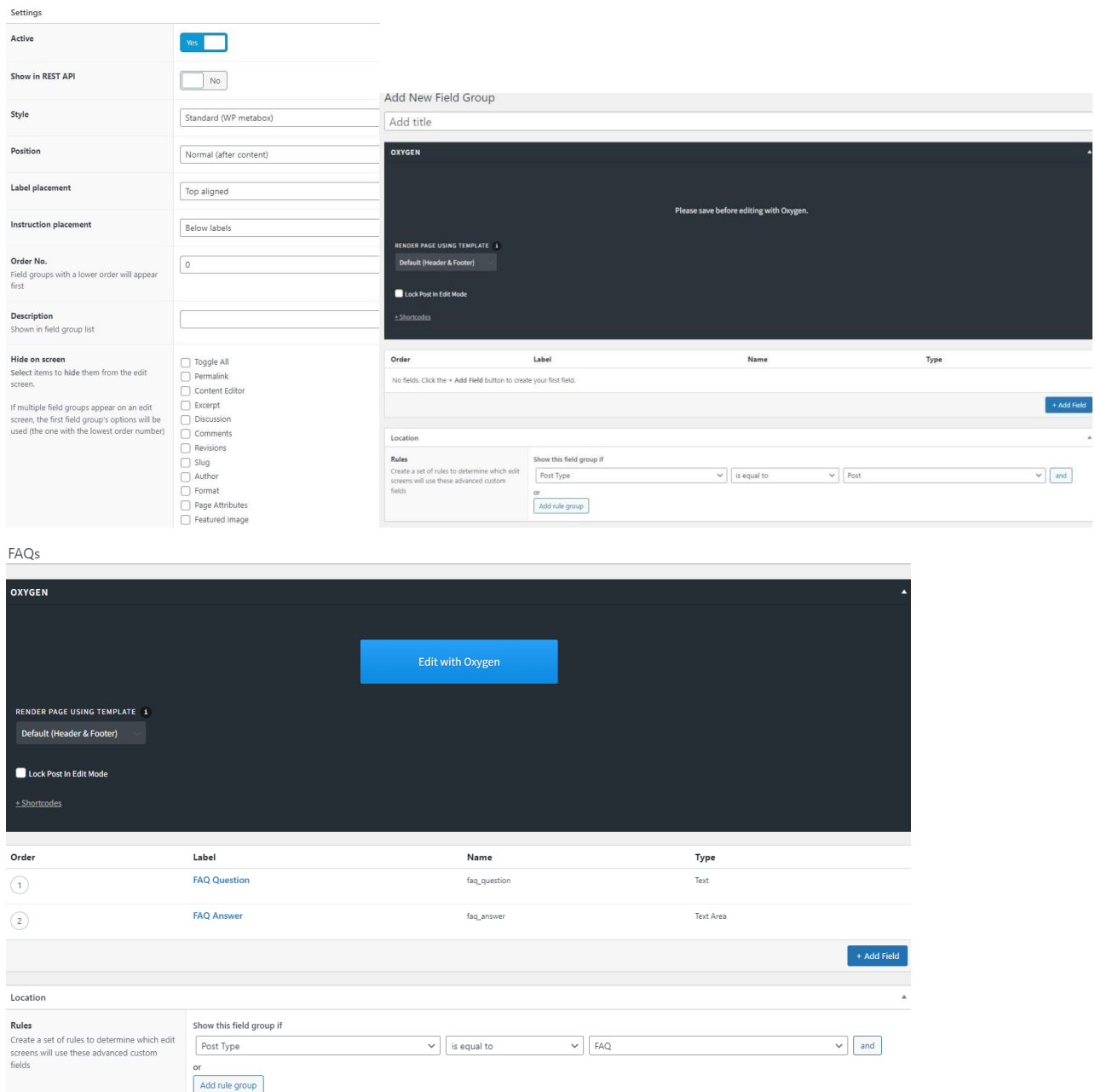


Рис 1. Елементи створення груп полів

3.2 Оксиджен.

Основна сторінка плагіну : <https://oxygenbuilder.com/>

Даний плагін є основним інструментом розробки дизайну проекту.

Оксиджен дозволяє створити певні шаблони сторінок і спрощує процес додавання та налаштування вигляду системи. Також даний плагін разом з розширеними спеціальними полями є хорошим інструментом розробки системи, який дозволяє швидко налаштувати як вигляд так і функціонал.

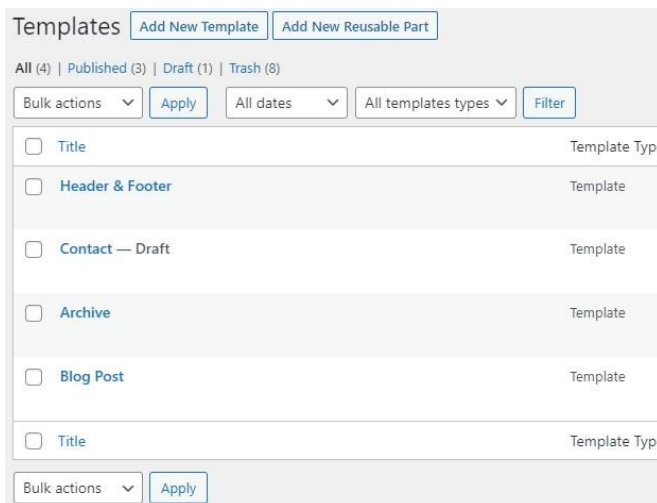
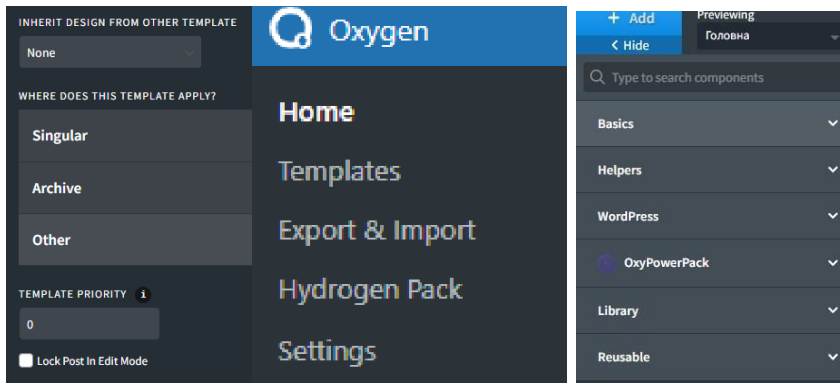


Рис 2 Поля налаштувань дизайну сторінок Оксиджен

3.3 Менеджер роботи

Основна сторінка плагіну : <https://wpjobmanager.com/>

Даний плагін містить базовий функціонал для пошуку роботи. Він був використаний для створення та відображення вакансій, редагування вакансій. Також даний плагін надає гнучке редагування та додавання певних змін до структури вакансії.

Jobs [Add New](#)

All (9) | Published (9)
English (0) | Ukrainian (9) | All languages (9)

Bulk actions All dates 9 items

<input type="checkbox"/>	Position	Type	Location	Posted	Expires	Categories	★	👤	Actions
<input type="checkbox"/>	.NET Engineer with Azure, Gambling Platform Data Art	Freelance	Odessa	January 20, 2022 by DevOps	March 25, 2022	Full Time / Remote	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	BI Architect epam	Full Time	Lviv	January 20, 2022 by DevOps	March 30, 2022	Full Time	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Java Engineer epam	Part Time	Anywhere	January 20, 2022 by DevOps	February 25, 2022	Internship	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Senior UI/UX Designer N-iX	Freelance	Dnipro	January 20, 2022 by DevOps	March 31, 2022	Full Time / Remote	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Lead QA Engineer Eleks	Full Time	Ternopil	January 17, 2022 by DevOps	February 28, 2022	Full Time / Remote	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Senior .Net Engineer N-iX	Full Time	Ivano-Frankivsk	January 17, 2022 by DevOps	February 28, 2022	Full Time / Remote	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Trainee .Net Backend Engineer GlobalLogic	Internship	Kyiv	January 16, 2022 by DevOps	February 28, 2022	Trainee	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Junior Test Automation Engineer SoftServe	Freelance	Kyiv	January 16, 2022 by DevOps	February 15, 2022	Full Time / Remote	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>
<input type="checkbox"/>	Junior Java Software Engineer SoftServe	Full Time	Lviv	January 16, 2022 by DevOps	January 31, 2022	Full Time	-	-	<input type="button" value="🔍"/> <input type="button" value="✎"/> <input type="button" value="🗑️"/>

Job categories

Add New Job category

Name

The name is how it appears on your site.

Slug

The "slug" is the URL-friendly version of the name. It is usually all lowercase and contains only letters, numbers, and hyphens.

Parent Job category

Assign a parent term to create a hierarchy. The term Jazz, for example, would be the parent of Bebop and Big Band.

Description

The description is not prominent by default; however, some themes may show it.

<input type="checkbox"/>	Name	Description	Slug
<input type="checkbox"/>	Full Time	—	full-time
<input type="checkbox"/>	Full Time / Remote	—	full-time-remote
<input type="checkbox"/>	Internship	—	internship
<input type="checkbox"/>	Part time	—	part-time
<input type="checkbox"/>	Trainee	—	trainee

Рис 3. WP Job Manager

3.4 Конструктор резюме

Основна сторінка плагіну : <https://wordpress.org/plugins/resume-builder/>

Даний плагін є хорошим інструментом для створення резюме.

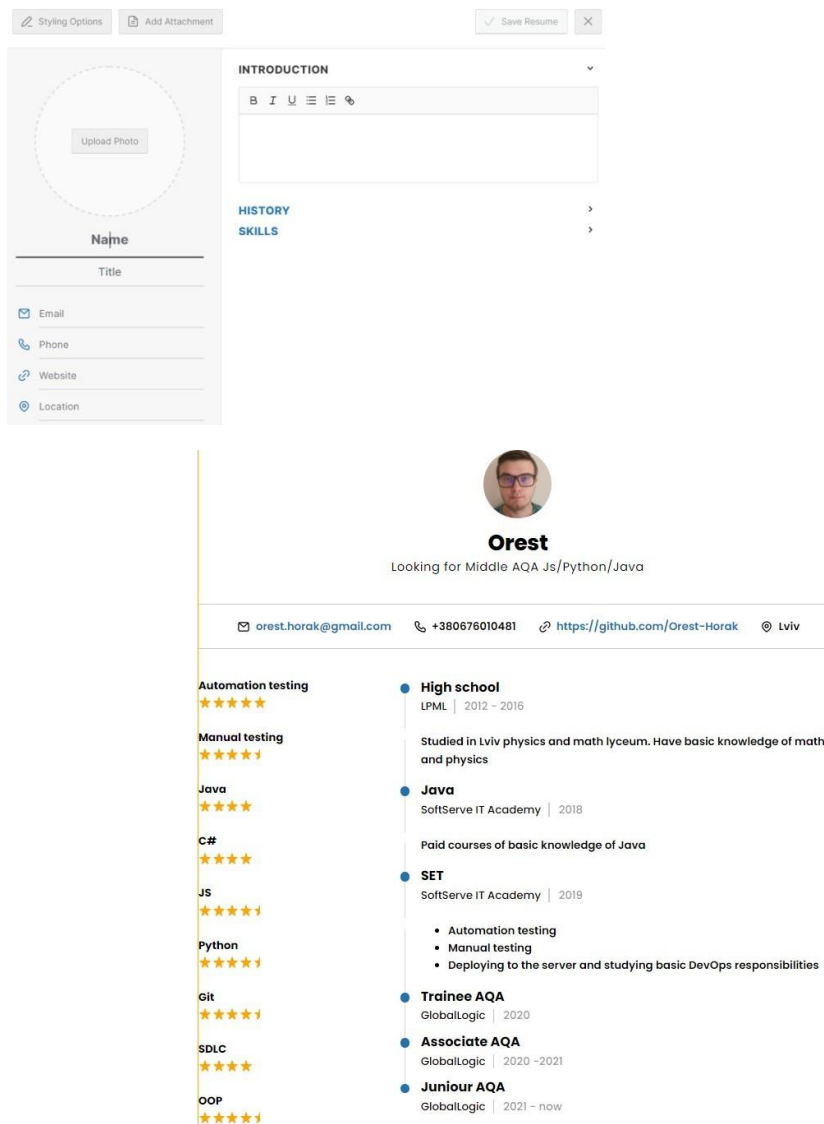


Рис. 4 Конструктор резюме

3.5 Багатомовність

Основна сторінка плагіну: <https://wpml.org/ru/>

Даний інструмент є зручним для створення багатомовних сайтів.

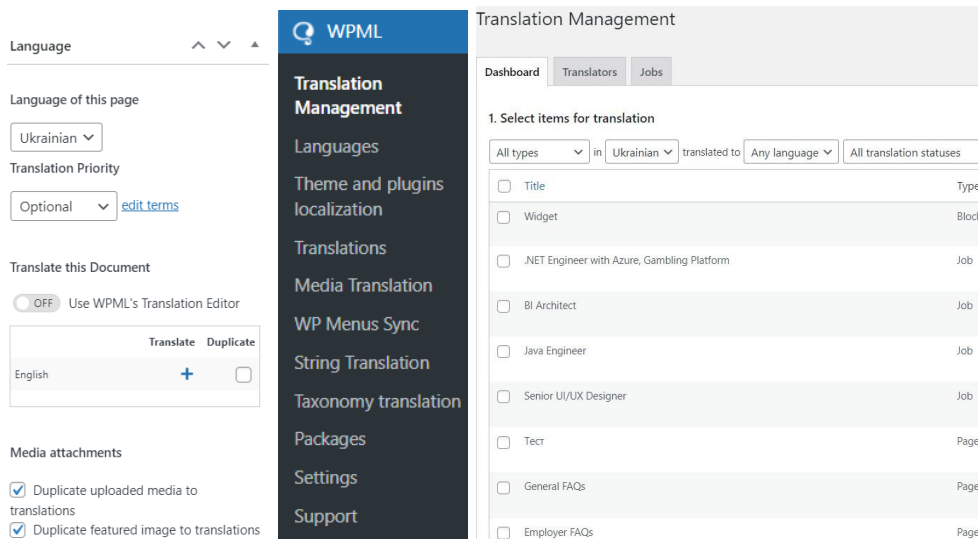


Рис. 5 WPML Multilingual

3.6 WP Cerber Security

Основна сторінка плагіну: <https://wpcerber.com/>

Даний плагін є чудовим інструментом для створення хорошого рівня захисту системи. Всі його функції є дуже корисними та зручними у налаштуванні та використанні.

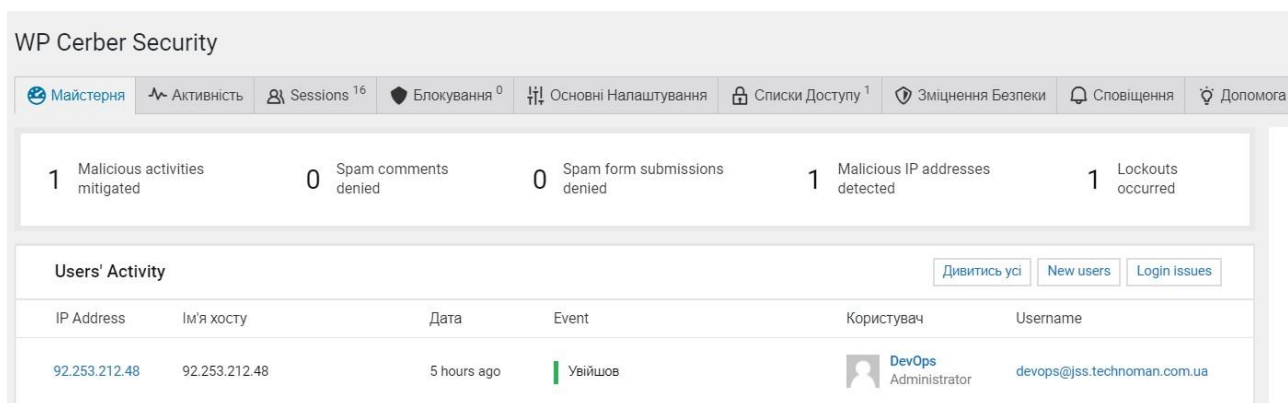


Рис. 6 Цербер

4. Двонаправлений зв'язок

4.1 Двонаправлений зв'язок загальні поняття. Фрагменти

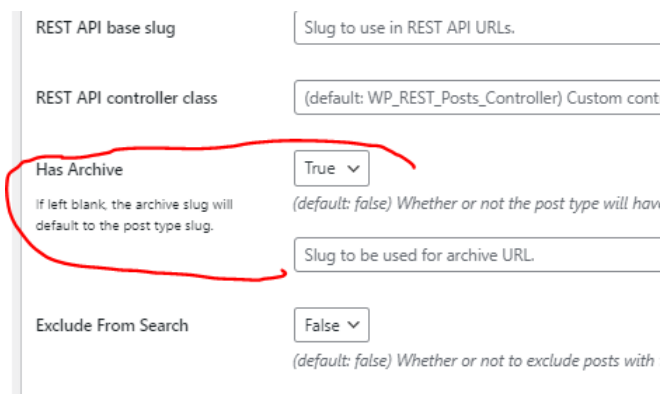
Вордпрес надає потужні механізми, вбудовані в ядро для структурування вмісту, а саме спеціальні типи публікацій і класифікацій, які можна використовувати для створення структурних шаблонів, але іноді цього недостатньо.

Хоча механізми двонаправленого зв'язку відкривають широкий спектр можливостей для структурування вмісту, є одне ключове співвідношення, яке вони не охоплюють, зокрема: багато до багатьох відносин.

Розглянемо конкретний приклад використання двонаправленого зв'язку в даній системі для вакансій та компаній.

Першим кроком потрібно створити спеціальну групу полів. Назвемо її *Company & Vacancy Field*. В цій групі полів додаємо поле *associated_with*

Окрім цього створюємо два типи постів *Company* та *Vacancy*. Також потрібно звернути увагу на те, щоб при створенні типу постів *Company* та *Vacancy* в налаштуваннях поле **Has Archive** було *true*.



REST API base slug

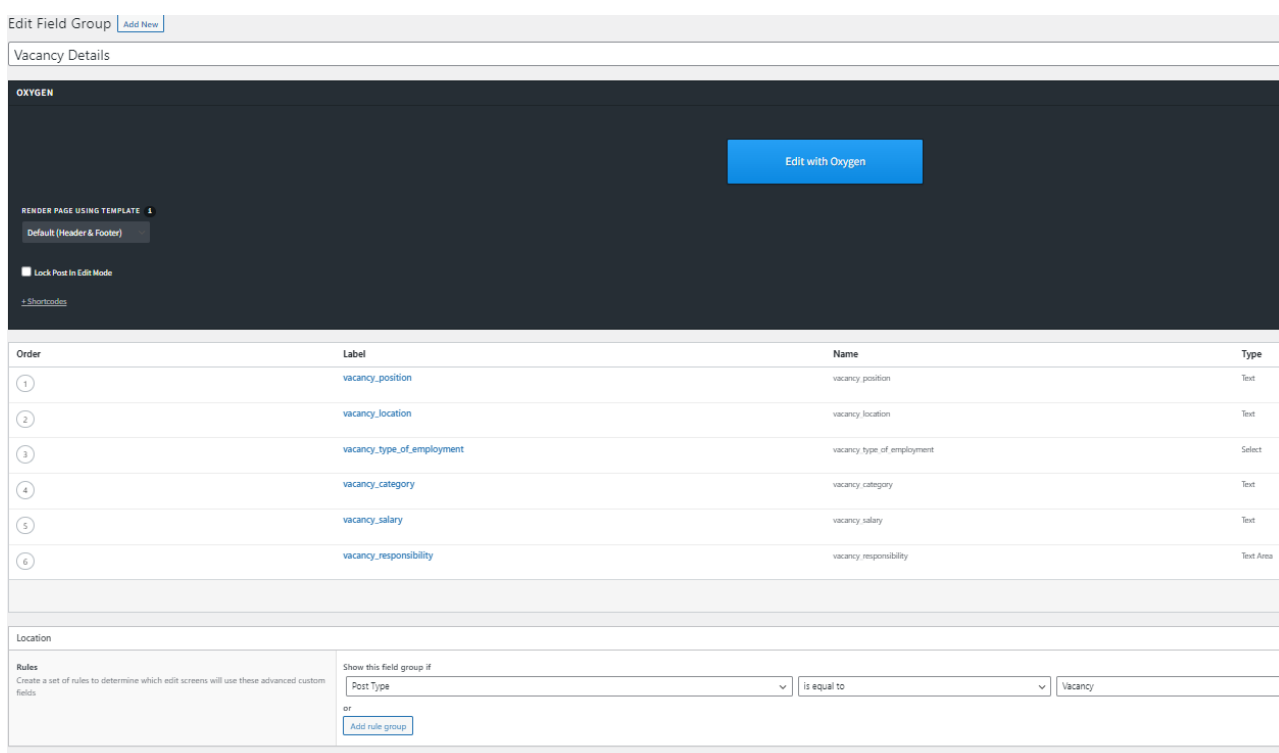
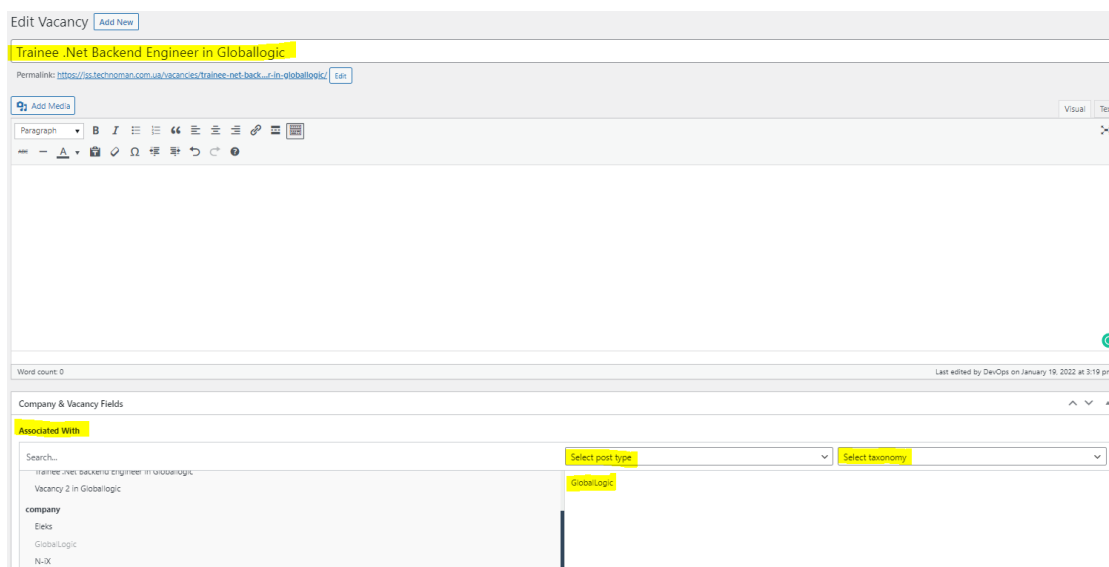
REST API controller class

Has Archive
If left blank, the archive slug will default to the post type slug. (default: false) Whether or not the post type will hav

Exclude From Search
(default: false) Whether or not to exclude posts with

The image shows two screenshots of the WordPress Admin interface. The top left screenshot is the 'Add/Edit Post Types' page for 'Vacancies'. It shows fields for 'Post Type Slug' (vacancies), 'Plural Label' (Vacancies), and 'Singular Label' (Vacancy). The top right screenshot is the 'Add/Edit Post Types' page for 'Companies', with 'Post Type Slug' (companies), 'Plural Label' (Companies), and 'Singular Label' (Company). A red circle highlights the 'Vacancies' and 'Companies' menu items in the right-hand sidebar. The bottom screenshot is the 'Edit Field Group' page for 'Company & Vacancy Fields'. It shows a table with one field: 'Associated With' (Label), 'associated_with' (Name), and 'Relationship' (Type). Below the table, there are rules for showing the field group if the 'Post Type' is 'Vacancy' or 'Post'.

Після цього потрібно створити ще дві групи полів для кожного типу постів (Company та Vacancy) для подальшого заповнення інформацією.



Останнім кроком налаштування двонаправленого зв'язку є додавання фрагменту зв'язків. Фрагмент коду можна взяти з

<https://www.advancedcustomfields.com/resources/bidirectional-relationships/> .

Потрібно тільки звернути увагу на останній фільтр, де ми маємо вказати ту назву, яку ми створили у групі полів *Company & Vacancy Field*. Тобто у нашому випадку потрібно змінити назву на *associated_with*

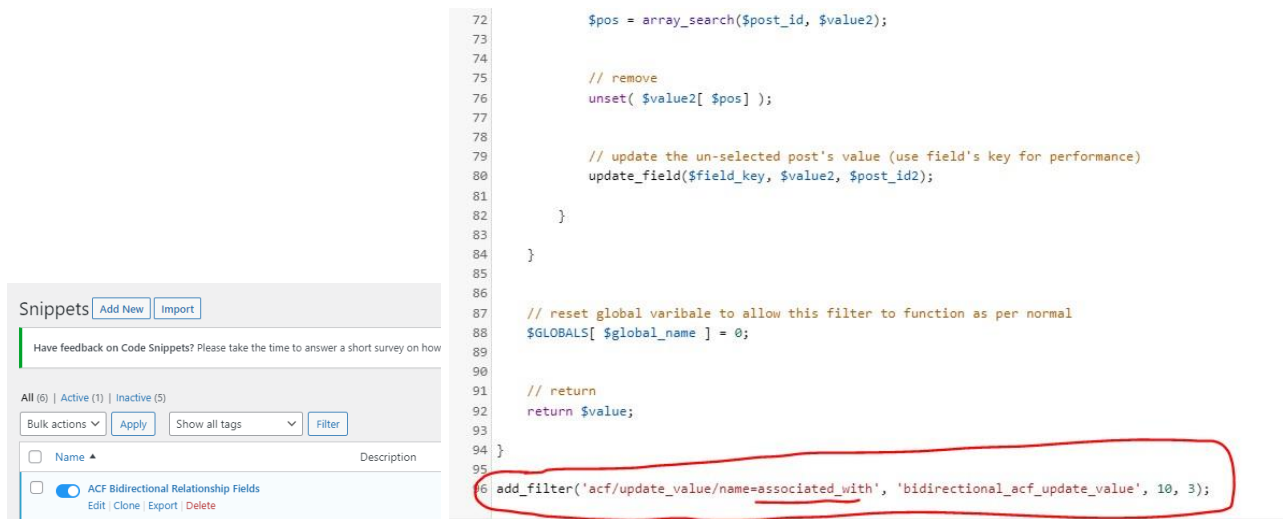


Рис. 7 відображення процесу налаштувань двосторонніх зв'язків

4.2 Динамічне відображення полів

Динамічні дані дозволяють зв'язувати елементи дизайну, створені в Оксиджен, з даними БД MySQL.

Наприклад, можна зв'язати елемент заголовка в шаблоні публікації в блозі з заголовком публікації в базі даних. Заголовок кожної публікації в блозі буде розміщено в елементі заголовка. Аналогічно, можна зв'язати текстовий елемент із вмістом публікації або зв'язати зображення чи фон розділу з пропонованим зображенням публікації.

Щоб вставити динамічні дані в будь-який текстовий елемент, потрібно двічі клацнути на текст, щоб відредагувати його.

Потім натиснути Вставити дані на верхній панелі інструментів, щоб відкрити

діалогове вікно вставки динамічних даних.

Певні властивості можна встановити за допомогою динамічних даних. Ці властивості мають кнопку даних праворуч від поля введення. Натисніть цю кнопку, щоб відкрити діалогове вікно Вставити динамічні дані.

Ця функція дозволяє робити такі дії, як установка URL-адреси фонового зображення розділу для пропонованого зображення або атрибута alt зображення для значення спеціального поля.

Insert Dynamic Data

Post

Title Content Excerpt Date Categories, Tags, Taxonomies Custom Field/Meta Options Comments Number Advanced Custom Field

Featured Image

Title Caption Alt

Author

Display Name Bio Meta / Custom Field

Current User

Display Name Bio Meta / Custom Field

Blog Info

Site Title Site Tagline Other

Archive

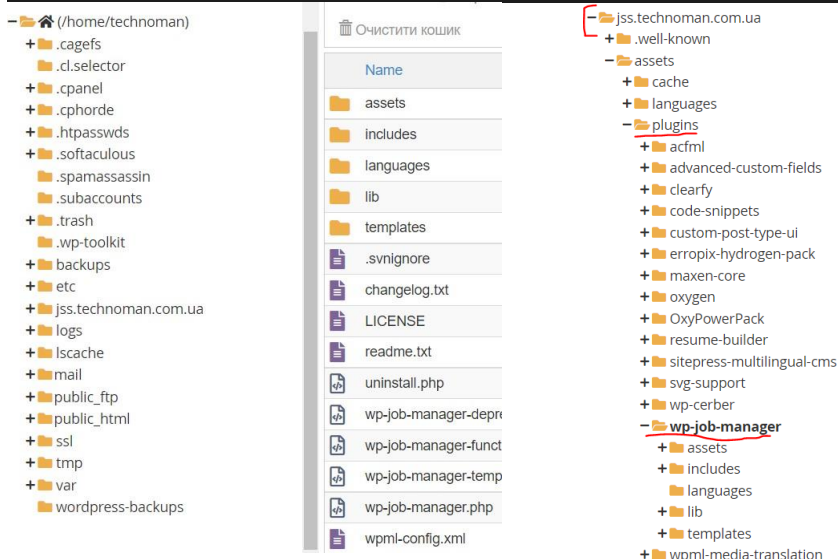
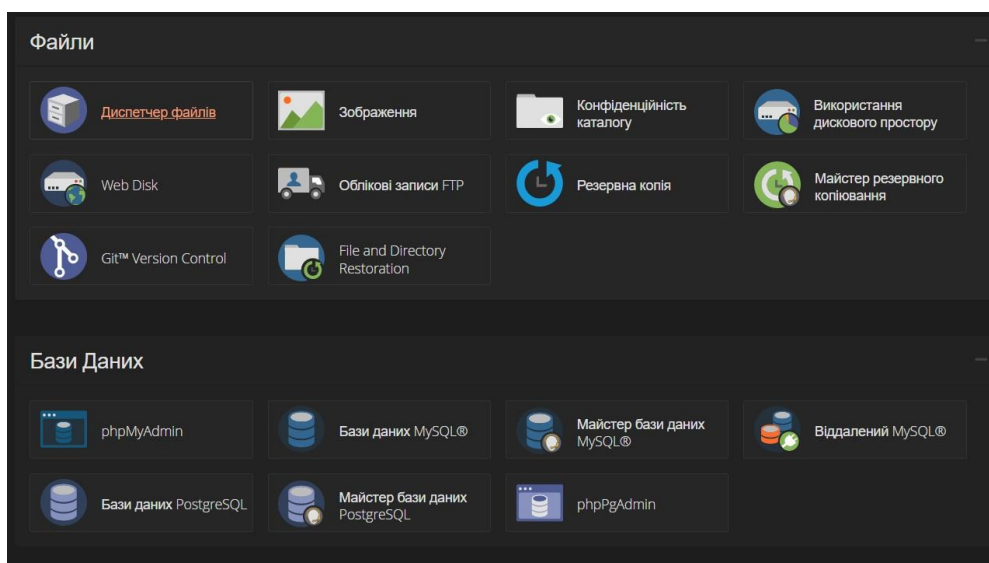
Archive Title Archive Description

Advanced

5. Редагування функціоналу плагінів в коді

5.1 Структура проекту. Сі-панел.

Всі інструменти системи розміщені на сі-панелі. Також в диспетчері файлів ми можемо доступитись до коду проекту та маємо змогу додати певні зміни до плагінів або інших частин системи, що надає більшу гнучкість в добавлянні нового функціоналу.



assets	4 KB	2 лист. 2021 р., 11:48	httpd/unix-direc
includes	4 KB	2 лист. 2021 р., 11:48	httpd/unix-direc
languages	4 KB	2 лист. 2021 р., 11:48	httpd/unix-direc
lib	4 KB	2 лист. 2021 р., 11:48	httpd/unix-direc
templates	4 KB	2 лист. 2021 р., 11:48	httpd/unix-direc
.svnignore	288	2 лист. 2021 р., 11:48	text/x-generic
changelog.txt	47,35 KB	2 лист. 2021 р., 11:48	text/plain
LICENSE	34,32 KB	2 лист. 2021 р., 11:48	text/x-generic
readme.txt	30,91 KB	2 лист. 2021 р., 11:48	text/plain
uninstall.php	1,12 KB	2 лист. 2021 р., 11:48	text/x-generic
wp-job-manager-deprecated.php	2,47 KB	2 лист. 2021 р., 11:48	text/x-generic
wp-job-manager-functions.php	45,77 KB	2 лист. 2021 р., 11:48	text/x-generic
wp-job-manager-template.php	33,45 KB	2 лист. 2021 р., 11:48	text/x-generic
wp-job-manager.php	1,74 KB	2 лист. 2021 р., 11:48	text/x-generic
wpml-config.xml	883 байт	2 лист. 2021 р., 11:48	text/x-generic

5.2 Доповнення плагінів

Доповнення існуючих плагінів новим функціоналом є досить важливим для повноцінної розробки системи. Розглянемо яким чином можливо відредагувати сам плагін та додати наприклад поле Salary.

Спочатку відкриваємо файл `functions.php` і додаємо поле `salary`

```
add_filter( 'submit_job_form_fields', 'frontend_add_salary_field' );
```

Потім пишемо функцію:

```
function frontend_add_salary_field( $fields ) {
    $fields['job']['job_salary'] = array(
        'label'      => __( 'Salary ($)', 'job_manager' ),
        'type'       => 'text',
        'required'   => true,
        'placeholder' => 'e.g. 20000',
        'priority'   => 7
    );
    return $fields;
}
```

Це додає текстове поле зарплати в нижній частині форми вакансії, яке має позначку «Зарплата», є обов'язковим, не має заповнювача та впорядковується на 7-й позиції.

Для того щоб відобразити поле salary є два варіанти:

- Ми можемо замінити шаблон для єдиного списку вакансій і вивести туди метадані за допомогою `get_post_meta()`.
- Ми можемо використовувати інший фільтр і відображати мета за допомогою нашої власної функції.

Спочатку нам потрібно знову підключити нашу функцію в правильне місце.

Цього разу ми використовуємо гачок дії:

```
add_action( 'single_job_listing_meta_end', 'display_job_salary_data' );
```

Потім ми пишемо функцію, яка отримує значення мета і виводить його у форматі списку:

```
function display_job_salary_data() {  
    global $post;  
  
    $salary = get_post_meta( $post->ID, '_job_salary', true );  
  
    if ( $salary ) {  
        echo '<li>' . __( 'Salary:' ) . ' $' . esc_html( $salary ) . '</li>';  
    }  
}
```

Можна додати нові фільтри до форми пошуку, додавши поле, а потім змінивши пошукові запити за допомогою фільтрів. У прикладі нижче додається спадне поле «зарплата», яке потім фільтрує списки на основі попередньо визначених діапазонів.

```
<?php  
/**  
 * This can either be done with a filter (below) or the field can be added directly to the job-filters.php template file!  
 *  
 * job-manager-filter class handling was added in v1.23.6  
 */  
add_action( 'job_manager_job_filters_search_jobs_end', 'filter_by_salary_field' );  
  
function filter_by_salary_field() {  
    ?>  
    <div class="search_categories">  
        <label for="search_categories"><?php _e( 'Salary', 'wp-job-manager' ); ?></label>  
        <select name="filter_by_salary" class="job-manager-filter">  
            <option value=""><?php _e( 'Any Salary', 'wp-job-manager' ); ?></option>  
            <option value="upto20"><?php _e( 'Up to $20,000', 'wp-job-manager' ); ?></option>  
            <option value="20000-40000"><?php _e( '$20,000 to $40,000', 'wp-job-manager' ); ?></option>  
            <option value="40000-60000"><?php _e( '$40,000 to $60,000', 'wp-job-manager' ); ?></option>  
            <option value="over60"><?php _e( '$60,000+', 'wp-job-manager' ); ?></option>  
        </select>  
    </div>  
    <?php  
}  
  
/**  
 * This code gets your posted field and modifies the job search query  
 */  
add_filter( 'job_manager_get_listings', 'filter_by_salary_field_query_args', 10, 2 );
```

```

function filter_by_salary_field_query_args( $query_args, $args ) {
    if ( isset( $_POST['form_data'] ) ) {
        parse_str( $_POST['form_data'], $form_data );

        // If this is set, we are filtering by salary
        if ( ! empty( $form_data['filter_by_salary'] ) ) {
            $selected_range = sanitize_text_field( $form_data['filter_by_salary'] );
            switch ( $selected_range ) {
                case 'upto20' :
                    $query_args['meta_query'][] = array(
                        'key'     => '_job_salary',
                        'value'   => '20000',
                        'compare' => '<',
                        'type'    => 'NUMERIC'
                    );
                    break;
                case 'over60' :
                    $query_args['meta_query'][] = array(
                        'key'     => '_job_salary',
                        'value'   => '60000',
                        'compare' => '>=',
                        'type'    => 'NUMERIC'
                    );
                    break;
                default :
                    $query_args['meta_query'][] = array(
                        'key'     => '_job_salary',
                        'value'   => array_map( 'absint', explode( '-', $selected_range ) ),
                        'compare' => 'BETWEEN',
                        'type'    => 'NUMERIC'
                    );
                    break;
            }

            // This will show the 'reset' link
            add_filter( 'job_manager_get_listings_custom_filter', '__return_true' );
        }
    }
    return $query_args;
}

```

Рис. 7 Приклад реалізації фільтрів

6. Розробка моделі даних для системи

6.1 Концептуальне проектування бази даних

Концептуальне проектування бази даних полягає у розбитті проекту на простіші задачі виходячи з поставленого завдання. На цьому етапі обираємо всі необхідні інструменти проектування та створюємо локальні концептуальні моделі даних. Було обрано реляційну модель даних та систему управління базами даних MySQL. Як результат, ми отримали такі відношення (див рис.1 зв 'язки в базі даних)

6.2 Логічне проектування бази даних

Під час логічного проектування бази даних, була створена схема з урахуванням обраної моделі представлення даних. Модель даних отримана в результаті концептуального проектування була взята за основу при створенні цієї схеми. В процесі розробки було відмежовано функціонал відповідно до ролей користувачів і як результат, утворились нові відношення.

6.3 Фізичне проектування бази даних

Після логічного проектування була визначена структура бази даних, а саме атрибутів, зв'язків та сутностей. Логічне та фізичне проектування досить тісно пов'язані тож будь які зміни задля підвищення продуктивності, які були внесені на етапі фізичного проектування, також вплинуть на структуру логічної моделі даних. У результаті проектувань ми отримали таку схему бази даних(див рис.1)

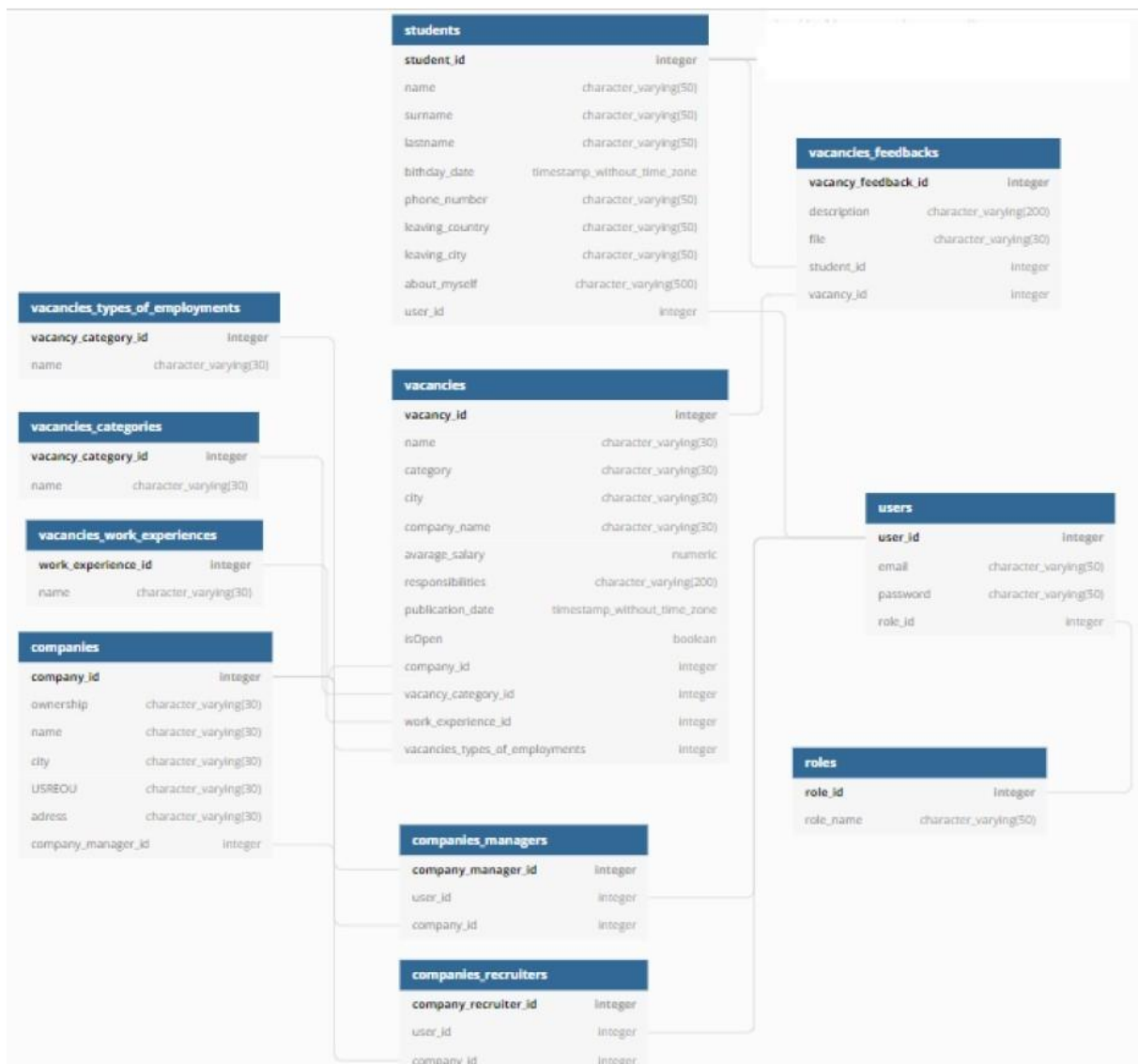


Рис 1. Зв'язки в базі даних

7. Результати реалізації системи пошуку роботи



Рис. 1 Головна сторінка

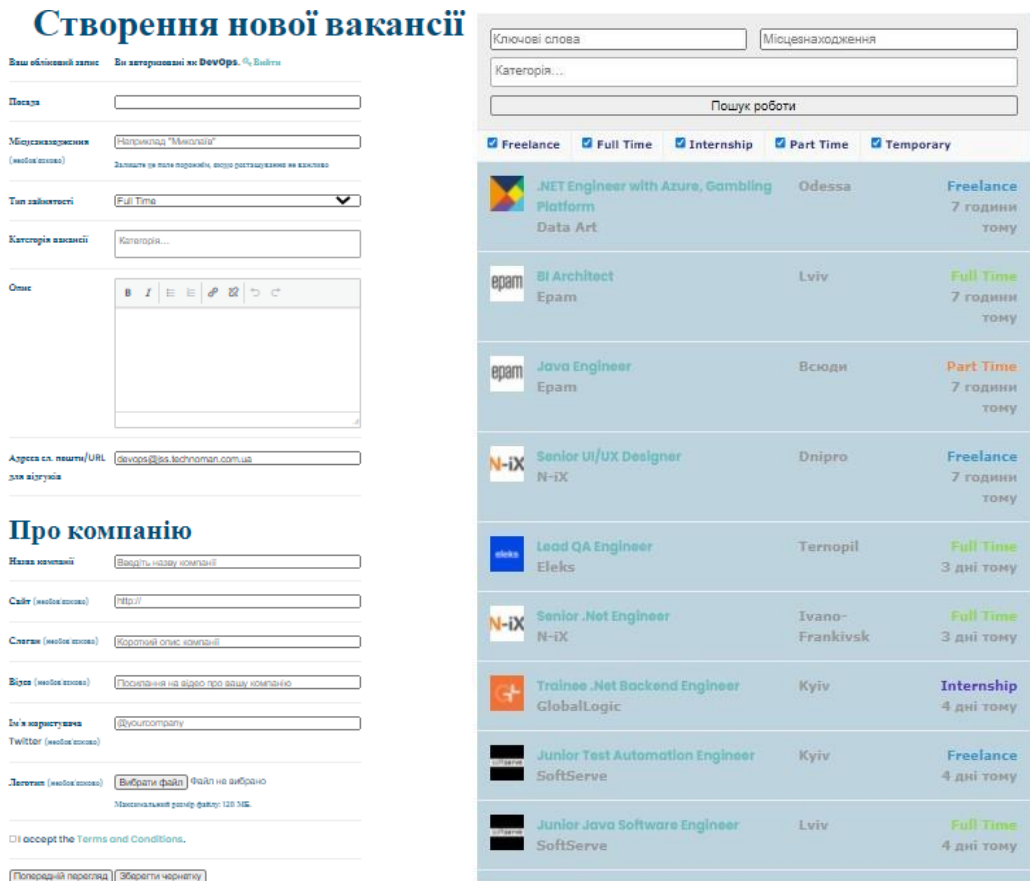


Рис. 2 Створення вакансії та список всіх вакансій з фільтром

Перелік опублікованих вакансій та терміни завершення їх актуальності

Ваші оголошення будуть показані в таблиці нижче.

Заголовок	Закрита?	Дата публікації	Термін дії вакансії закінчується
.NET Engineer with Azure, Gambling Platform	-	20 Січня, 2022	25 Березня, 2022
BI Architect	-	20 Січня, 2022	30 Березня, 2022
Java Engineer Відмітити закритою • Створити дублікат • Надати	-	20 Січня, 2022	25 Лютого, 2022
Senior UI/UX Designer	-	20 Січня, 2022	31 Березня, 2022
Lead QA Engineer	-	17 Січня, 2022	28 Лютого, 2022
Senior .Net Engineer	-	17 Січня, 2022	28 Лютого, 2022
Trainee .Net Backend Engineer	-	16 Січня, 2022	28 Лютого, 2022
Junior Test Automation Engineer	-	16 Січня, 2022	15 Лютого, 2022
Junior Java Software Engineer	-	16 Січня, 2022	31 Січня, 2022

Рис. 3 Менеджер вакансій

FAQS для студентів



Даний розділ має за собою на меті донести інформацію як користуватися сайтом для студентів. Також в цьому розділі можна знайти відповіді на найпопулярніші питання.

- **Student question 8**
Student answer 8
- **Student question 7**
Student answer 7
- **Student question 6**
Student answer 6
- **Student question 5**
Student answer 5
- **Student question 4**
Student answer 4
- **Student question 3**
Student answer 3 something 3 something 3 something 3
- **Student question 2**
Student answer 2 something 2 something 2 something 2
- **Third FAQ**
3 FAQ
- **This is sample question**
Yes sure it is

Рис. 4 Запитання користувачів

8. Висновок

У ході розробки системи пошуку роботи, був проведений аналіз вимог та пошук схожих аналогів і формування загального уявлення вигляду системи та проблем, які можуть виникнути в ході роботи. Також було спроектовано вигляд цієї системи і визначено певний функціонал, який потрібно реалізувати. Під час створення сторінок та реалізації певного функціоналу за допомогою розширених спеціальних полів виникли певні труднощі з утворенням двосторонніх зв'язків. Сама система розміщена на сервері та нею можна керувати через сі-панел. Також через цей інструмент можливо зібрати корисну статистику системи, зокрема її пропускну здатність та інші корисні дані. За допомогою Оксиджен були розроблені певні шаблони сторінок та подальше наповнення всією необхідною інформацією. Також були використані плагіни менеджер роботи і конструктор резюме для часткового покриття функціоналу. Окрім цього, доповнення плагінів допомогло краще зрозуміти структуру проекту. В даній системі також реалізована можливість багатомовних сторінок, а саме автоматичного перекладу(за потреби). Також однією із важливих частин цієї системи є налаштування безпеки за допомогою Цербер. Цей плагін допоміг досить добре налаштувати систему безпеки, окрім цього можна відслідкувати всю активність у системі та покращити засоби безпеки.

Використані джерела:

1. URL: <https://wpjobmanager.com/document/tutorial-adding-a-salary-field-for-jobs/>
2. WP Cerber: <https://wpcerber.com/getting-started/>
3. ACF: <https://www.advancedcustomfields.com/>
4. Oxygen: <https://oxygenbuilder.com/>
5. WP Job Manager: <https://wpjobmanager.com/>
6. Resume Builder: <https://wordpress.org/plugins/resume-builder/>
7. WPML: <https://wpml.org/ru/>
8. Snippet: <https://wpcute.ru/blog/что-такое-snippet-y-i-kak-ih-ispolzovat-v-wordpress/>
9. Bi-directional relationship: <https://sixtwo.tech/bidirectional-post-relationships-in-wordpress/>
10. Bi-directional relationship:
<https://www.advancedcustomfields.com/resources/bidirectional-relationships/>
11. Тестовий сайт : <https://jss.technoman.com.ua/>
12. Карпова Т.С. Базы данных: модели, разработка, реализация [Текст] / Т.С. Карпова - СПб: Питер, 2002. - 303с.
13. Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. [Текст]: учебник/ Т. Коннолли, К. Бегг ; -2-е изд., испр. и доп. - М.: Изд. дом "Вильямс": 2001. - 1111с.
14. Jason Couleman, Brian Messenlener Building Web Apps with WordPress as an Application Framework 2nd Edition