

ДИПЛОМНА РОБОТА

ДР.Пс – 12.00.000 ПЗ

Група Пс-2015

Підлісецький А. В.

2019

Кафедра Інформаційних технологій та програмної інженерії

УДК 004

ДИПЛОМНА РОБОТА

Тема Розробка інформаційного веб-ресурсу реабілітаційного центру АТО «Нові Крила»

Напрямок підготовки 6.050103

(код і назва спеціальності)

ПОЯСНЮВАЛЬНА ЗАПИСКА

ДР.ПІс – 15.00.000 ПЗ

(позначення)

Студент

Підлісецький А.В.

(підпис) (дата) (розшифрування підпису)

Керівник проекту

к.т.н.

Ващишак С.П.

(посада) (підпис) (дата) (розшифрування підпису)

Нормоконтроль

к.т.н.

Андрейко В.М.

(посада) (підпис) (дата) (розшифрування підпису)

Допускається до захисту

Завідувач кафедри

д.т.н., доц.

Мельничук С.І.

(посада) (підпис) (дата) (розшифрування підпису)

ПВНЗ УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА

Факультет Інформаційних технологій
Кафедра Інформаційних технологій та програмної інженерії
Напрямок підготовки 6.050103 – «Програмна інженерія»

ЗАТВЕРДЖУЮ:
Завідувач кафедри ІТПІ
С.І. Мельничук
“ ” 2019 р.

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ)**

Студенту Підлісецькому Андрію Васильовичу

1. Тема проекту (роботи) Розробка інформаційного веб-ресурсу реабілітаційного центру для воїнів АТО «Нові Крила»

Затверджена наказом ректора Університету Короля Данила від 15.11.2018 р. №20/4

2. Термін задачі студентом закінченого проекту (роботи) 10.06.2019

3. Вихідні дані до проекту (роботи) Python Django, SQLite база даних, Bootstrap

4. Зміст пояснювальної записки (перелік питань, що їх належить розробити)
1. Завдання веб-сайту опис аналогів і використані технології 2. Розробка взаємозв'язків між функціоналом 3. Розробка структури проекту та його функціоналу.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Тема роботи, мета, задачі досліджень. Робота з замовником. Структура інтерфейсу веб-ресурсу. Адміністративна панель Сторінка «Зворотного зв'язку». Висновки.

6. Консультанти з проекту (роботи), із зазначенням розділів проекту, що стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1			
2			
3			

7. Дата видачі завдання 30.10.2018

Керівник _____ Ващишак.С.П.

Завдання прийняв до виконання _____ Підлісецький.А.В.

КАЛЕНДАРНИЙ ПЛАН

Пор №	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1.	Завдання веб-сайту опис аналогів і використані технології		
2.	Розробка взаємозв'язків між функціоналом		
3.	Розробка допоміжних функцій та інтерфейсів адмін-панелі		
4.	Розробка структури проекту та його функціоналу		
5.	Оформлення графічного матеріалу та підготовка до захисту роботи		

Студент-дипломник _____ Підлісецький А. В.
(підпис) (розшифровка підпису)

Керівник проекту _____ Ващишак.С.П.
(підпис) (розшифровка підпису)

АНОТАЦІЯ

Під час виконання дипломної роботи було розроблено інформаційний веб ресурс для реабілітаційного центру воїнів АТО «Нові Крила» . Даний ресурс розроблений для спрощення комунікації між персоналом центру і його пацієнтами ,інформування ветеранів АТО\ООС про надання реабілітації а також залучення додаткових коштів для подальшого функціонування центру.Для власників була розроблена адміністративна панель управління сайтом.

SUMMARY

During the implementation of the thesis, an informational web resource was developed for the rehabilitation center of the soldiers ATO «New Wings». This resource is designed to simplify the communication between staff of the center and its patients, inform veterans ATO \ OOS about the provision of rehabilitation, as well as attract additional funds for the further operation of the center. For owners, an administrative control panel for the site was developed.

РЕФЕРАТ

Розрахунково-пояснювальна записка: 68 сторінки, 13 рисунків, 2 додатки.

Об'єктом дослідження є центр реабілітації учасників АТО та членів їх сімей «Нові крила».

Метою роботи є дослідження процесу формування веб-представництва у мережі Інтернет для інформування учасників АТО про надання медичних послуг центром реабілітації, виявлення недуг і проблем психологічного або фізичного характеру, та створення інформаційного веб-ресурсу для спрощення зв'язків між центром і пацієнтами та центром і спонсорами.

Відповідно до поставленого завдання в роботі проводиться дослідження процесу формування зв'язків між учасниками АТО та центром реабілітації, що існували до недавнього часу. Для полегшення даного процесу, економії часу, підвищення якості виявлення проблем у пацієнтів, формування напрямків реабілітації і залучення відповідних фахівців розроблено інформаційни веб-ресурс для воїнів АТО «Нові крила».

Реабілітаційний центр, веб-ресурс, реабілітація, пацієнт, АТО, Django, Python, Bootstrap, Html.

ЗМІСТ

ЗМІСТ	6
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ.....	7
ВСТУП	8
1 ЗАВДАННЯ ВЕБ-сайту, опис аналогів і використані технології.....	9
1.1 Нормативно правова база з питань регулювання діяльності в мережі Інтернет	9
1.2 Опис веб-сайтів аналогічного призначення	10
1.4 Робота з замовником.....	15
2 РОЗРОБКА ВЗАЄМОЗВ'ЯЗКІВ МІЖ ФУНКЦІОНАЛОМ САЙТУ.....	18
2.1 Створення моделей і міграція.....	18
2.2 Робота з гіпер текстовою розміткою.....	23
2.3 Опис бази даних	28
3 РОЗРОБКА СТРУКТУРИ ПРОЕКТУ ТА ЙОГО ФУНКЦІОНАЛУ	29
3.1 Розробка головної сторінки інформаційного ресурсу.....	29
3.2 Розробка сторінки зворотного зв'язку	35
3.3 Адміністративна панель і її налаштування	39
ВИСНОВКИ.....	45
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	46
Додаток А.....	48
Додаток Б	54

						ДР.Пс – 15.00.000 ПЗ		
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>						
<i>Розроб.</i>		<i>Підлісецький А.В.</i>			<i>Розробка програмного забезпечення для формування розкладу занять студентів Пояснювальна записка</i>	<i>Літ.</i>	<i>Ар.</i>	<i>Акрушієв</i>
<i>Перевір.</i>		<i>Вацшиак С.П.</i>					6	70
<i>Реценз.</i>						<i>УКД, Пс – 2015</i>		
<i>Н. Контр.</i>		<i>Андрейко В.М</i>						
<i>Затверд.</i>		<i>Мельничук С.І</i>						

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

CSS - Cascading Style Sheets

HTML - HyperText Markup Language

JS - JavaScript

					ДР.Шс – 24.00.000 ПЗ	Арк.
						7
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підп.</i>	<i>Дата</i>		

ВСТУП

Актуальність теми. В час масової інформатизації суспільства наявність власного веб-представництва в мережі інтернет є ознакою успішності. Наявність власного веб ресурсу є не просто необхідністю, але спрощує і прискорює надання послуг інформуючи клієнтів про надання послуг. Також наявність веб сайту спрощує зв'язок з пацієнтами допомагає дізнатися про їх не дуги в пару кліків.

Інтернет-благодійність щодня набуває все більшої популярності у світ разом з динамікою росту даного засобу комунікації. Якщо ще 10-15 років тому основним засобом комунікації благодійних організацій зі своїми потенційними донорами були поштові листи, то сучасні інтернет-технології активно займають провідні позиції. А тому кожна благодійна організація має на власному веб-сайті кнопку, за допомогою якої кожна людина через інтернет може безпечно переказати на рахунок цієї організації кошти за для розв'язання тієї чи іншої проблеми. Доступність інформації зручний інтерфейс все для того щоб спростити процес інтеграції в мирне життя. Після повернення з війни не всі можуть вжитися в світі де не стріляють. В наслідок важкої соціально економічної ситуації адабтуватися до мирного життя вдається не кожному. І націй сумній ноті появляються волонтери.

Об'єктом дослідження є центр реабілітації учасників АТО та членів їх сімей «Нові крила».

Метою роботи є дослідження процесу формування веб-представництва у мережі Інтернет для інформування учасників АТО про надання медичних послуг центром реабілітації, виявлення недуг і проблем психологічного або фізичного характеру, та створення інформаційного веб-ресурсу для спрощення звязків між центром і пацієнтами та центром і спонсорами.

Методи дослідження – загальна теорія інформації, математична статистика, методи створення веб – сайтів.

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		8

1 ЗАВДАННЯ ВЕБ-САЙТУ, ОПИС АНАЛОГІВ І ВИКОРИСТАНІ ТЕХНОЛОГІЇ

В загальному, реабілітаційні веб-сайти використовуються для:

- виявлення людей, яким потрібна допомога;
- спрощення надання реабілітаційними закладами різноманітних послуг;
- допомоги працівникам реабілітаційних організацій знаходити контакт з клієнтами;
- залучення коштів спонсорів для забезпечення функціонування реабілітаційних центрів тощо.

1.1 нормативно правова база з питань регулювання діяльності в мережі Інтернет

В Україні юридичні норми що стосуються регулювання діяльності в мережі Інтернет можна знайти у загальних законах. Один із них це Закон України «Про доступ до публічної інформації» [1], в якому описано: засади опублікування і поширення інформації в мережі Інтернет її розпорядниками, права на доступ користувачів до інформації тощо. Загальні правила і гарантії інформаційно-правових відносин встановлює закон України «Про Інформацію» [2]. Згідно з цим законом Інтернет є універсальним засобом комунікації та способом передачі, оприлюднення і поширення інформації. Більшість норм закону стосуються цифрових прав.

Закон «Про авторське право» [3] містить наступні визначення:

- «Веб-сайт - сукупність даних, електронної (цифрової) інформації, інших об'єктів авторського права і (або) суміжних прав тощо, пов'язаних між собою і структурованих у межах адреси веб-сайту і (або) облікового запису

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		9

власника цього веб-сайту, доступ до яких здійснюється через адресу мережі Інтернет, що може складатися з доменного імені, записів про каталоги або виклики і (або) числової адреси за Інтернет-протоколом.»

- Веб-сторінка – складова частина веб-сайту, що може містити данні, електронну(цифрову) інформацію, інші об’єкти авторського права і (або) суміжних прав тощо;

- власник веб-сайту – це особа яка володіє веб ресурсом та встановлює умови використання сайту;

- Гіперпосилання - сформульована відповідно до стандартів адреса веб-сайту або веб-сторінки, що здійснює адресацію на веб сайт або його сторінку;

- Постачальник послуг хостингу – особа, яка надає ресурси та послуги власникам сайту для їх розміщення в мережі Інтернет та забезпечення доступу.

1.2 опис веб-сайтів аналогічного призначення

У сучасному світі веб-сайт є ознакою не тільки прозорості перед суспільством, а також ознакою успішності організації .Web-сайт – це своєрідний інтерфейс між організацією та її партнерами і клієнтами. Тому створення сайту є одним з головних завдань організації в наш час.

Інтернет-благодійність щодня набуває все більшої популярності у світ разом з динамікою росту даного засобу комунікації . Якщо ще 10-15 років тому основним засобом комунікації благодійних організацій зі своїми потенційними донорами були поштові листи, то сучасні інтернет-технології активно займають провідні позиції. А тому кожна благодійна організація має на власному веб-сайті кнопку, за допомогою якої кожна людина через інтернет може безпечно переказати на рахунок цієї організації кошти за для

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		10

розв'язання тієї чи іншої проблеми. Доступність інформації зручний інтерфейс все для того щоб спростити процес інтеграції в мирне життя.

Revived Soldiers Ukraine – це благодійна Американська організація яка допомагає із реабілітацією українських поранених у Лос-Анджелесі (рис. 1.1), а також займається збором коштів. Веб-сайт цієї організації складається з логотипу горизонтального меню яке містить посилання на сторінки [4]:

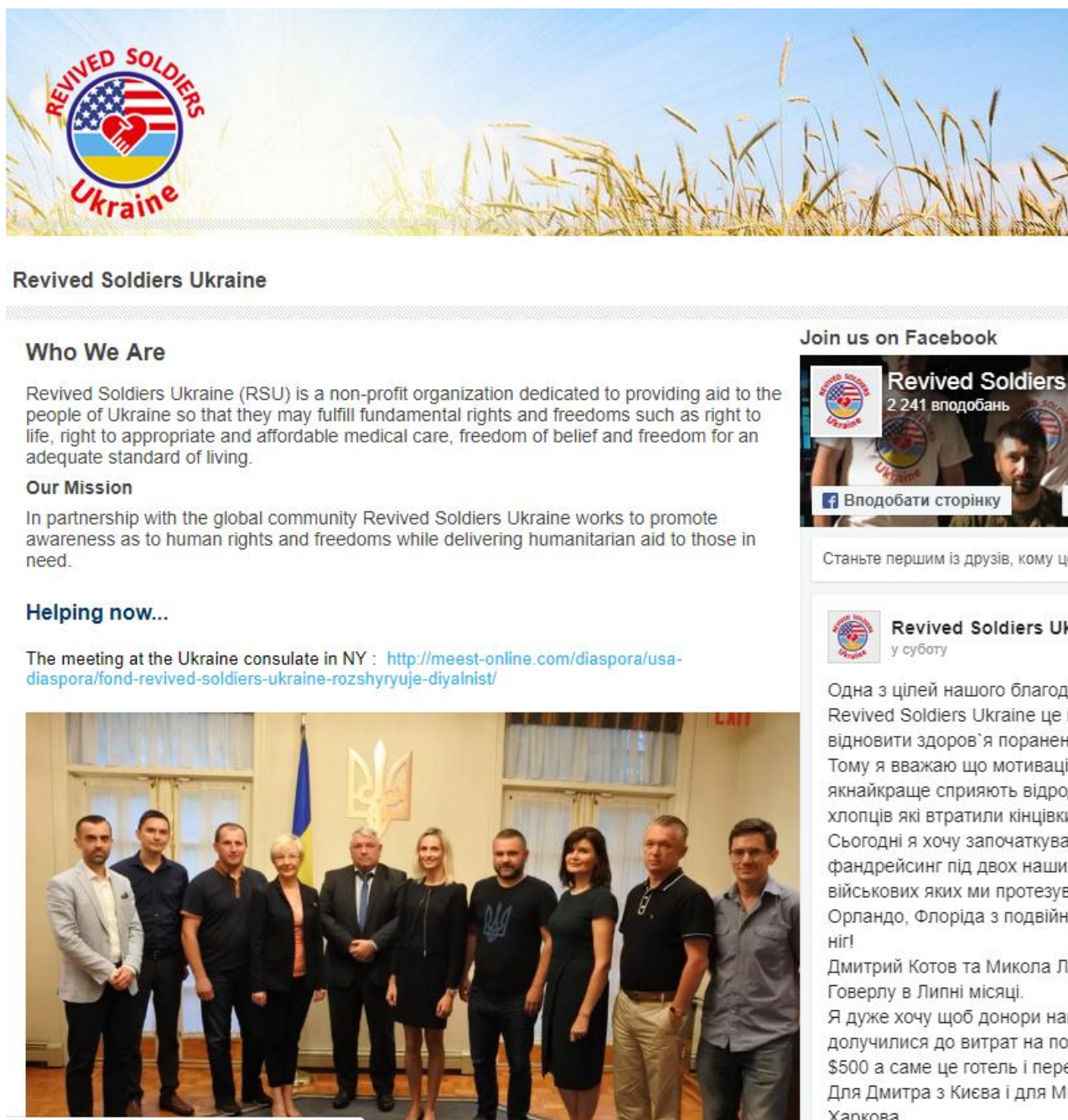


Рисунок 1.1 – Сайт організації Revived Soldiers Ukraine

									Арк.
									11
Змн.	Арк.	№ докум.	Підп.	Дата	ДР.Шс – 11.00.000 ПЗ				

- сторінку яка містить інформацію про організацію її діяльність;
- звітність хто скільки пожертвував і кому ця допомога була надана;
- способи співпраці;
- сторінка подій.

Важливим компонентом є кнопка швидкого переказу коштів яка доступна майже на кожній сторінці сайту. Але варто зазначити, що є в цього сайту і мінуси - один із них він не налаштований на дисплей мобільних телефонів.

1.3 Використанні технології при розробці сайту

Одним із найважливіших етапів розробки будь-якого сайту є вибір технології на якому він буде написаний. Варто приділити цьому процесу достатньо часу проаналізувати які технології справляться з поставленими задачами. Також не варто вибирати ті технології які не є популярними або вважаються «мертвими» [5-8]. При виборі технологій варто враховувати потенційну аудиторію, розмір проекту і його тип. Для сайтів візиток можна використовувати CMS. Для сайтів середнього рівня складності, таких як портали національного масштабу використовують фреймворки [9-12]. Більш важчі проекти, наприклад соціальні мережі, розробляються на чистій мові програмування.

При виборі технологій варто звертати увагу і на документацію це дозволить швидше розв'язувати необхідні задачі документації зазвичай можна знайти готове рішення, що рятує розробника від створення «велосипеду». Не варто забувати і про вартість проекту, оплату хостингу, а також використання різних платних сервісів - все це слід детально обрахувати. Безпека відіграє дуже важливу роль в роботі сайту. Конфіденційність стає більшою, якщо робота сайту буде пов'язана з

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		12

переказами кошів. Отже, провівши детальний аналіз можна зекономити свій час і гроші.

Для розробки веб-сайту реабілітаційного центру Нові Крила було використано наступні технології:

- Bootstrap – це безкоштовний набір інструментів з відкритим кодом, призначений для створення веб-сайтів та веб-додатків, який містить шаблони CSS та HTML для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу, а також додаткові розширення JavaScript;

Python - інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією;

Django — високорівневий відкритий Python-фреймворк для розробки веб-систем. Якщо ви фанат запального класичного джазу то даний фреймворк саме для вас оскільки він названий на честь відомого в світі джазмена Джанго Рейнхардта. Архітектура Django подібна на модель MVC «модель-вид-контролер» (рис.1.2) [13- 14].

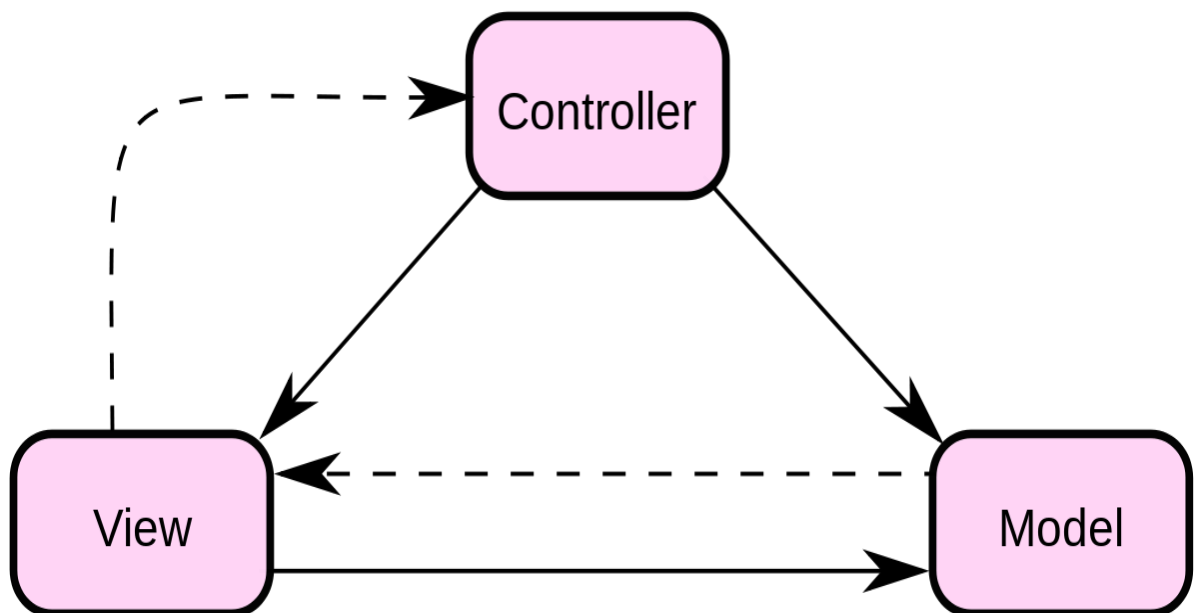


Рисунок 1.2 - Архітектурний шаблон MVC

Однак розробники внесли зміни до даної моделі і те що в звичайній MVC називається контролер отримало назву вид, а те що називається вид

називається шаблоном таким чином ми отримали MTV ні це не улюблений музичний телеканал одного з розробників, а всього лиш «модель-темплейт-вид».

В Django спростили процес створення адміністративної частини веб сайту оскільки в фреймворку є вбудована адміністративна панель, а також можливість робити різні маніпуляції з контентом веб-сайту [15].

pip — система керування пакунками, яка використовується для встановлення та управління програмними пакетами, які написані на Python .дана система прискорює установку пакетів,бібліотек, а також вона здатна встановити фреймворки. Окрім встановлення pip виконує ще функції оновлення та видалення [16];

PyCharm — інтегроване середовище розробки для мови програмування Python. Його перевагами є зручний інтерфейс завдяки якому робота з кодом пришвидшується в двічі якщо порівняти даний редактор коду з іншими то для оновлення він не потребує встановлення додаткових плагінів все що потрібно користувач отримує при встановленні даного продукту. Також важливу роль грає і функціонал в який входить спеціальна консоль, в якій можна проводити різноманітні операції (рис.1.3) [16].

```
Terminal: Local x +
TypeError: __init__() takes 1 positional argument but 2 positional ar

(venv) C:\mysite>python manage.py makemigrations
No changes detected

(venv) C:\mysite>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, new_wings, session
Running migrations:
  No migrations to apply.

(venv) C:\mysite>python manage.py runserver
```

Рисунок 1.3 – Приклад запуску сервера через консоль

Adobe Photo Shop – графічний редактор який призначений для редагування цифрових фотографій. Його особливості полягають в великій кількості інструментів для створення і обробки зображення, зручність у використанні. Завдяки функції «layers» накладати на зображення текст геометричні фігури тощо [17].

MySQL – вільна система керування реляційними базами даних була створена як альтернатива комерційним системам. Враховуючи те що MySQL підтримується більшістю мовами програмування її застосовують для створення динамічних сторінок [18];

Git – розподілена система керування версіями файлів та спільної роботи. Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні і злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів [17];

Virtualenv – програма для створення і управління оточеннями Python. Дозволяє створити середовище зі своїми окремими модулями, налаштуваннями і програмами. Середовище обмежується рамками одного каталогу. Програма є дуже зручна для роботи з різними версіями одних і тих же модулів, для створення проектів, у яких "все з собою", які не залежать від операційної системи [19].

1.4 робота з замовником

Для успішної роботи над будь-яким проектом в галузі ІТ необхідно не тільки правильно виконувати поставлене завдання, але і важливу роль в цьому процесі відіграє спілкування з клієнтом. Головне в співпраці з клієнтом - це задавати як більше чітких питань, наприклад: “Що ви очікуєте?

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		15

”Яке призначення буде мати майбутній продукт?” - це одне з головних правил роботи з клієнтом, чим чіткіше обговорити всі нюанси з клієнтом тим менше в подальшому буде проблем з розробкою.

Друге правило – це ініціатива: - розробнику потрібно брати ініціативу в свої руки; генерування хороших ідей і приємних для клієнта рішень приведе до того, що клієнт буде погоджуватись з розробником, а кожен з нас знає, що поганий замовник це той, який говорить що робити і не хоче сприймати іншої думки.

Оскільки бувають випадки, коли клієнт може не знати, що саме повинно бути на веб сайті, то тут при годяться дизайнерські вміння розробника, а саме - вміння малювати. Мова йде про скетчінг. Поняття пішло від англійського слова sketch як швидкий малюнок, який допомагає відобразити головну ідею і задум дизайнера.

Лояльність буде третім складовим. Важливо не ігнорувати правки клієнта, потрібно завоювати його довіру. Якщо виконати два попередніх правила, то з довірою буде все впорядку. Варто пам’ятати, що клієнт завжди правий і робота видеться для нього “не він прийшов до вас, а ви до нього”. В першу чергу потрібно надати якісну послугу клієнту за його гроші. Розробник має розуміти, що в першу чергу, він повинен зробити те, що від нього вимагають, а не те, що йому підказує досвід.

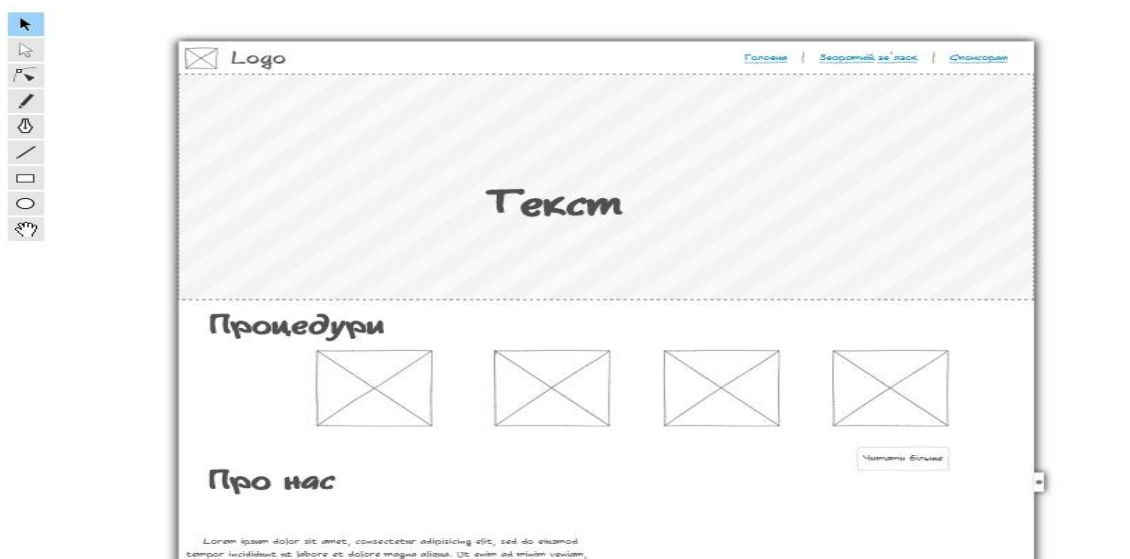


Рисунок 1.4 - Створенні макети майбутнього сайту

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		16

Четверте правило – якщо клієнт прийшов до розробника з завданням, з яким він не знайомий, не рекомендовано відмовлятися від роботи. З цієї ситуації існує два виходи: прийняти замовлення і навчитися чогось новому або делегувати клієнта до спеціаліста який вміє і знає як вирішити ту чи іншу проблему.

І останнє не менш головне, потрібно робити роботу як для себе, навіть якщо замовлення є не цікавим потрібно виконати його якісно. У моєму випадку, під час розмови з замовником було зображено швидкий малюнок (рис. 1.4), який відобразив все що повинно було міститися на майбутньому веб сайті.

Для створення даного скетчу було застосовано онлайн сервіс NinjaMock [21] для протипування інтерфейсів за допомогою цього додатку можна швидко і з максимальною точністю спланувати приблизний функціонал.

1.5 постановка завдання на дипломне проектування

Виходячи з завдання у дипломній роботі необхідно було створити інформаційний веб-ресурс для реабілітаційного центру з ручним дизайном і зрозумілим контентом. А також:

- форму зворотнього зв'язку як дозволяє повідомити контактні данні інформацію про недугу вибрати тип реабілітації;
- сторінку з інформацією для меценатів і спонсорів яка містить в собі спеціальну кнопку для переказу коштів, а також логотипи організацій спонсорів;
- створити зручний інтерфейс адміністратора завдяки якому можна швидко редагувати контент (видаляти/додавати фотографії, заміна тексту тощо).

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		17

2 РОЗРОБКА ВЗАЄМОЗВ'ЯЗКІВ МІЖ ФУНКЦІОНАЛОМ САЙТУ

2.1 створення моделей і міграція

Модель - це основне джерело даних .Вони містять в собі поля і поведінку даних. По суті її можна назвати структурою бази даних. Всі моделі унаслідуються від базового класу Model і містять поля визначень [22].

Під час розробки інформаційного веб-ресурсу мною було створено 9 моделей:

```
class Slider(models.Model):
    name = models.CharField(blank=True, max_length=50)
    screen = models.ImageField(blank=True, null=True,
upload_to='image')
    def __str__(self):
        return 'Слайд'.format(self.id)
class Information(models.Model):
    image = models.ImageField(blank=True, null=True,
upload_to='image')
    text = models.CharField(max_length=500)
    def __str__(self):
        return 'Про нас'.format(self.id)
class Contact(models.Model):
    phone = models.CharField(max_length=150)
    email = models.EmailField(max_length=254)

class Social(models.Model):
    link = models.URLField(max_length=200)
    icon = models.ImageField(blank=False, null=True,
upload_to='image')
class Reabil(models.Model):
    text = models.TextField(max_length=1000)
    image = models.ImageField(blank=True, null=True,
upload_to='image')
```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		18

```

class Donate(models.Model):
    account = models.CharField(max_length=100)

class Sponsor(models.Model):
    logo = models.ImageField(blank=True, null=True,
upload_to='image')

class New(models.Model):
    title = models.CharField("Заголовок", max_length=100)
    images = models.ImageField(blank=False, null=True,
upload_to='image/')
    text = models.TextField("Опис")
    created = models.DateTimeField("Дата створення",
auto_now_add=True)
    keywords = models.CharField("Ключові слова", max_length=50)

    def __str__(self):
        return self.title

class Suport(models.Model):
    full_name = models.CharField("Прізвище Ім'я По батькові",
max_length=150)
    age = models.IntegerField
("ваш
вік", default=1, validators=[MaxValueValidator(10), MinValueValidat
or(1)])
    city = models.TextField("Місце проживання", max_length=200)
    problem = models.TextField("Проблема що торбує")
    live_through = models.TextField("Перенесині травми, операції
а також наяві захворювання: ", max_length=200)
    recent_treatment = models.CharField(max_length=200)
    rehabil_type = models.CharField("Вид реабілітації",
max_length=100, )
    exemption = models.BooleanField("Наявність пільг", )

    def __str__(self):
        return self.full_name

```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		19

В своїй роботі я використав наступні типи полів:

- `ImageField` – дозволяє завантажувати фотографії спадковує всі атрибути та методи з `FileField`, але також перевіряє, що завантажений об'єкт є дійсним зображенням.

- `CharField` – тип поля для зберігання рядків . Поле даного типу використовується разом з атрибутом `max_length` який встановлює скільки символів міститиме поле;

- `TextField` – є аналогічним типом поля як і `CharField` .Застосовується до більшості кількості символів;

- `EmailField` – це теж поле `CharField` яке використовує модуль `EmailValidator` який перевіряє правильність написання електронної пошти.

- `URLField` – поле яке використовує `URLValidator` для перевірки написання адреси.

- `BooleanField` – логічне поле яке може містити `True` або `False` віджет за замовчуванням даного поля `CheckboxInput`.

Після створення моделей потрібно активувати нашу модель, на що в Django є спеціальні команди такі як:

- `makemigrations` – дана команда застосовується для внесення змін, а також для того щоб зберегти їх ;

- `migrate` – команда яка здійснює всі не виконані міграції і оновлює базу даних.

Всі ці команди містяться в базовому модулі `manage.py`. Процес міграції відбувається наступним чином після створення всіх моделей я викликаю команду `python manage.py makemigrations` (рис.2.1) для того щоб зберегти всі моделі.

Після цього я використовую команду `python manage.py migrate` (рис.2.2), яка оновлює базу даних і синхронізує її згідно створених моделей.

					ДР.ІІс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		20

```
Microsoft Windows [Version 6.3.9600]
(c) Корпорация Майкрософт (Microsoft Corporation), 2013. Все права защищены.

C:\mysite>python manage.py makemigrations
```

Рисунок 2.1 - Команда для створення нової міграції

Потім моделі будуть проскановані і порівняні з версією, яка міститься в файлах міграцій, після чого будуть створені нові міграції.

Розглянемо цей процес детальніше: візьмемо вище наведений клас Slider і розберемо його:

class Slider - це ім'я майбутньої таблиці бази даних яка створиться в подальшому під час міграції. В дужках ми вказуємо модулі, в нашому випадку, це models. Під час написання моделі потрібно дотримуватися симантичних правил написання. Наприклад, назва класу завжди повинна складатися з великої букви, а також, якщо назва класу складається з двох слів то її варто записувати в CamelCase.

В class Slider входять наступні поля:

- name - це так зване поле нашої таблиці. Тобто, при створенні поля нам потрібно вказати його тип, а також деякі поля мають обов'язкові параметри, без яких створення таблиці не є можливим в CharField (Текстове поле). CharField потребує обов'язкового параметру max_length, в якому потрібно вказати яка кількість символів буде міститися у нашому полі;

- image – в ній заданий тип поля, який дозволяє працювати з графічними об'єктами, а точніше - завантажувати їх;

Також, про що треба сказати, це те, що в кожній таблиці після проведення міграції створюється додаткове поле для ідентифікатора.

За наведеним алгоритмом ми створили модель – Slider і відповідну міграцію. Далі нам потрібно підтвердити наші зміни в базі даних і, тим самим, завершити процес міграції. Для цього використовуємо команду migrate (рис.2.3). За допомогою даної команди можна як підтверджувати

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		21

зміни так і повертатися до колишніх міграцій. В загальному ж команда migrate може робити міграцію для всіх підключених додатків, повертатися до тієї чи іншої міграції, а також здійснювати міграцію для окремого додатку.

Отже, з врахуванням вищенаведеного, процес міграції відбувається в два етапи. Спочатку синхронізуються додатки без міграцій, а потім застосовуються міграції, які ще не використовувалися у базі даних.

Міграції зберігаються в так званих «файлах міграції» [23]. Це звичайні Python файли з класом міграції, які дотримуються певного інтерфейсу. У ньому декларативно можна описати всі необхідні операції та інше.

Базовий файл міграції моєї моделі Slider виглядає наступним чином (рис.2.2).

```
from django.db import migrations, models

class Migration(migrations.Migration):

    dependencies = [
        ('new_wings', '0008_auto_20190502_1939'),
    ]

    operations = [
        migrations.AddField(
            model_name='slider',
            name='screen2',
            field=models.ImageField(blank=True, null=True, upload_to='img'),
        ),
    ]
```

Рисунок 2.2 - Вигляд базового файлу міграції

При запуску файлу міграції (імпортування як модуля Python) Django шукає дочірний клас django.db.migrations.migration, який повинен називатися Migration. він аналізує чотири атрибути класу, які зазвичай використовуються тільки два:

- dependencies - це список залежних міграцій;
- Operations - список підкласів Operation, які визначають необхідні для міграції операції. Список операцій - найголовніше в міграції. Він декларативно описує необхідні операції для зміни структури бази даних.

Django аналізує всі операції і в пам'яті, будує схему всіх необхідних змін, щоб згенерувати необхідні SQL - запити. Схема в пам'яті також використовується щоб визначити відмінності між нашими поточними моделями і станом моделей, описаному в міграціях. Django проходить по всіх змінах і створює структуру моделей на момент останнього виконання makemigrations. Потім ця структура порівнюється зі структурою моделей в models.py і визначаються зміни, які ми внесли в наші моделі.

2.2 робота з гіпер текстовою розміткою

Html — це не мова програмування, це мова розмітки, яка вказує вашому браузеру, як відобразити вміст веб-сторінки, яку ми переглядаємо. Вона може бути простою чи складною, залежно від бажання веб-дизайнера, який її створює. Html складається з серії (елементів), які використовуються для вміщення, або «загортання» в них різних частин контенту, щоб вони відображалися чи діяли в певний спосіб. Ці елементи за допомогою початкових і кінцевих (тегів) можуть зробити слово чи зображення посиланням на будь-яку іншу сторінку, можуть відобразити текст курсивом, збільшити чи зменшити шрифт тощо.

Написання HTML документу не буде можливим без наступних тегів:

<!DOCTYPE> - призначені для визначення типу поточного документа - DTD (визначення типу документа, опис типу документа). Це необхідно для того, щоб браузер розумів, як слід інтерпретувати поточну веб-сторінку, оскільки HTML існує в декількох версіях, а саме: XHTML (EXtensible HyperText Markup Language, розширення мовного розміру гіпертексту), схожий на HTML, але відрізняється від нього по синтаксису. Для перегляду «не запускається» і розуміється, відповідно до вимог стандарту відображається веб-сторінка і необхідно в першу чергу кодувати питання[24];

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		23

<html> повідомляє браузеру, що це HTML-документ.

Тег <html> є контейнером, який містить в собі весь вміст веб-сторінки, включаючи теги <head> та <body>, крім тега <!DOCTYPE>.

Як правило, тег <html> йде в документі другим, після елемента, що визначає тип документа (Document Type Definition, DTD), що встановлюється через елемент<!DOCTYPE>[25].

<head> це частина, яка не відображається у веб-браузері під час завантаження сторінки. Він містить таку інформацію, як назва сторінки <title>, посилання на файл стилів CSS (якщо ви хочете стилізувати ваш HTML контент за допомогою CSS), посилання на власні іконки та інші метадані (дані про HTML, наприклад, хто його написав, та важливі ключові слова, які описують документ). Даний контейнер містить наступні теги:

- <title> - є обов'язковим тегом тексту, що розміщується в середині даного тегу і буде відображатися в стоці заголовку веб-браузера. Довжина тексту обмежується 60 символами, також текст повинен максимально описувати вміст веб-сайту [26];

- <meta> - за допомогою даних тегів можна створити опис сторінки і ключові слова для кращого пошуку в пошукових системах. Також тут можна створити автора і багато інших мета даних. Даний тег має свої атрибути серед яких найголовніше значення має charset даний тег дозволяє вказати кодування сторінки [27];

- <link> - даний тег дозволяє підключати додаткові модулі, а також саме цим тегом підключаються стилі [28].

Елемент <body> призначений для зберігання змісту веб-сторінки (контенту), що відображається у вікні браузера. Інформацію, яку ми виводимо в документі, слід розташовувати саме всередині контейнера <body>. До такої інформації належить текст, зображення, теги, скрипти JavaScript тощо. Тег <body> також застосовується для визначення кольорів посилань і тексту на веб-сторінці. Подібна практика в HTML 5 засуджується і натомість для вказівки колірної схеми рекомендується використовувати

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		24

стилі, застосовуючи їх до селектора BODY. Тим не менше, більшість атрибутів досі підтримується різними браузерами.

Елемент `<script>` дозволяє приєднувати до документа різні сценарії та має закриваючий тег. Обов'язковий при цьому текст сценарію може розташовуватися або всередині цього елемента, або в зовнішньому файлі. Якщо текст сценарію розташований в зовнішньому файлі, то він підключається за допомогою атрибутів елемента. Для елемента доступні атрибути `async`, `charset`, `defer`, `src`, `type`, а також глобальні атрибути. Зрозвитком технологій почали появлятися спеціальні CSS фреймворки ,які покликані пришвидчити створення сайтів .На сьогодні існують такі фреймфорки:

- Bootstrap;
- Skeleton;
- Semantic UI;
- UIKit;
- і багато інших.

Дані фреймворки спрощують створення HTML сторінки.Фреймворки використовують як початківці так і досвідчені розробники сайтів.

Я зупинив свій вибір на фрейм ворку Bootstrap не тільки через те що він є простим і популярним ,ще одною його відмінністю є те що він дозволяє робити зручний інтерактивний дизайн чого не може UIKit.

Для того щоб працювати з Bootstrap потрібно в контейнері `<head>` через тег `<link>` підключити стилі від даного фреймворку . Наступним кроком підключив бібліотеку скриптів . Код наведено нижче.

```
<meta content="width=device-width, initial-scale=1.0"
name="viewport">
<link crossorigin="anonymous"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/boo
tstrap.min.css" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T
" rel="stylesheet"><script
```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		25

```
src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.m
in.js"></script>
<script crossorigin="anonymous" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo
"src="https://code.jquery.com/jquery-
3.3.1.slim.min.js"></script>
<script crossorigin="anonymous" integrity="sha384-
UO2eT0CpHqdSJJQ6hJty5KVphtPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1
"src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/um
d/popper.min.js"></script>
<script crossorigin="anonymous" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEeAfff/nJGzIxFDsf4x0xIM+B07jRM
"
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/boots
trap.min.js">
</script>
```

Принцип розробки розмітки полягає в наступному для початку потрібно створити блочний елемент (div) з класом .container . даний клас робить контейнер фіксованим .Також в Bootstrap є клас .container-fluid який розтягує контейнер на всю ширину екрану .Для прикладу внизу я навів приклад коду:

```
<div class="row d-flex justify-content-center">
  <div class="col-md-10 text-align-left mt-3">
    {% for information in informations %}
      {{information.text}}
    {% endfor%}
  </div>
</div>
```

Варто зазначити що Bootstrap дозволяє створювати власну сітку за допомогою класів .row і .column вказується як пуду розташовуватися контент на сторінці чи в рядок чи в стовпчик. Далі потрібно створити колонку яка позначається класом .col. Максимум в одному рядку може бути 12 колонок. Даним колонкам можна давати розмір для цього потрібно до класу col дописати через тире цифру від 1 до 12 . Таким способом можна міняти колір

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		26

тексту ,давати відступи . Bootstrap підтримує технологію flex-box.Дана технологія покликана швидко вирівняти елемент по горизонталі і вертикалі. Для цього контейнеру в рядку класу треба прописати d-flex. Так ми оголошуємо що даний елемент є flex-елементом далі нам потрібно визначити як нам потрібно вирівнювати наш елемент за замовчуванням буде встановлено значення row тобто по горизонталі.Далі можна займатися переміщенням даного елемента дану функцію покладено на клас justify-content . Він дозволяє вирівнювати елементи по лівому краю(justify-content-start), по середині(justify-content-center) а також по правому краю(.justify-content-end) . Використовуючи всі ці функції я швидко розробив розмітку для проекту (рис.2.3).

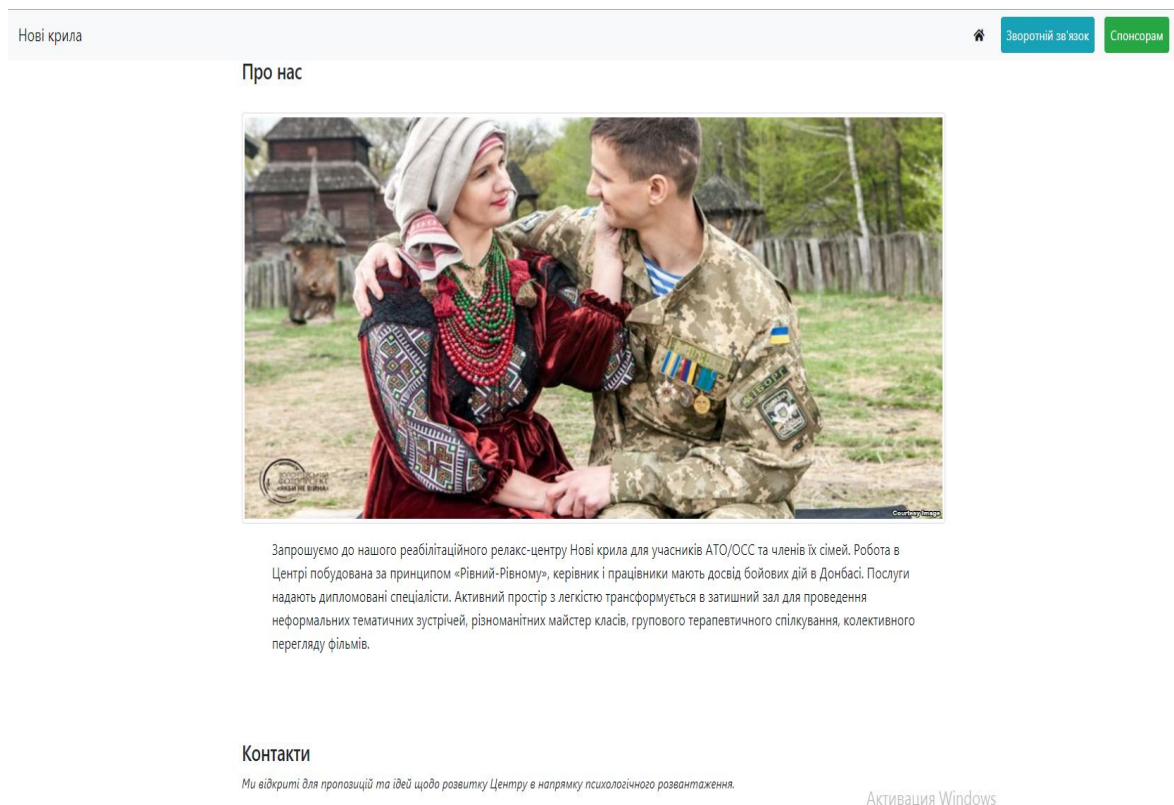


Рисунок – 2.3 – Вигляд головної сторінки сайту

Приклад коду головної сторінки можна переглянути в додатку можна переглянути в додатку (Додаток А).

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		27

2.3 опис бази даних

Для створення даного продукту я використав базу SqlLite оскільки вона є простою в використанні її рушій ,графічний інтерфейс ,також важливим фактором є те що дана база даних дозволяє зберігати всі данні в одному файлі що економить як час так і пам'ять. По спілкувавшись з колегами я зупинив свій вибір саме на цій базі даних [29].

SqlLite вона має наступні переваги:

- Вільна ліцензія;
- Безпека;
- Висока швидкість простих операцій;
- Економічна в плані ресурсів;
- тощо.

Також при виборі бази даних я оцінював і недоліки в Sql Lite вони наступні:

- Відсутність системи користувачів;
- Відсутність збільшення продуктивності.

В рахувавши ці фактори а також поспілкувавшись з колегами я вибрав саме цю базу даних. Sqllite варто використовувати для сайтів з низьким і середнім трафіком [30].

Окрім Django використовують разом з:

- Java
- PHP
- Ruby on Rails
- Trac

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		28

3 РОЗРОБКА СТРУКТУРИ ПРОЕКТУ ТА ЙОГО ФУНКЦІОНАЛУ

3.1 розробка головної сторінки інформаційного ресурсу

Головна сторінка, або домашня сторінка, це перша сторінка на яку заходить відвідувач при потраплянні на сайт. Розробка головної сторінки відбувається за більший час, ніж створення інших сторінок. При розробці домашньої сторінки я поділив її на п'ять умовних секцій (рис.3.1):

- Шапка сайту або Header яка містить меню навігації і логотип. Шапка – це один з головних елементів сайту який впливає на його популярність і зручність керуванні в цілому. Навігаційне меню, що міститься в ній дозволяє перейти на такі сторінки як «Спонсорам», «Зворотній зв'язок» а також кнопку яка дозволяє повернутися на головну:

```
<nav class="navbar sticky-top navbar-expand-lg navbar-light
bg-light ">
  <button aria-controls="navbarText" aria-expanded="false"
aria-label="Toggle navigation" class="navbar-toggler border-
0"
      data-target="#navbarText" data-toggle="collapse"
type="button">
    <span class="navbar-toggler-icon "></span>
  </button>
  <a class="navbar-brand" href="#">Нові крила</a>
  <div class="collapse navbar-collapse" id="navbarText">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item active">
        <a class="nav-link btn" href="{% url 'home'
%}"><i class="fas fa-home"></i><span class="sr-
only">(current)</span></a>
      </li>
      <li class="nav-item col-md">
```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		29

```

        <a class="nav-link text-center text-dark"
href="{% url 'support' %}">Зворотній зв'язок</a>
    </li>
    <li class="nav-item ml-3">
        <a class="nav-link btn btn-success text-
white" href="{% url 'donate' %}">Спонсорам</a>
    </li>
</ul>
<span class="navbar-text">
</span>
</div>

</nav>

```

- Слайдер містить в собі фотографії мотивуючого змісту власник сайту може додавати до п'яти фотографій через адміністративну панель:

```

<div class="carousel slide" data-ride="carousel"
id="carouselExampleControls">
    <div class="carousel-inner">
        {% for slider in sliders %}
        <div class="carousel-item {% if forloop.first %}
active{% endif %}">
            
        </div>
        {% endfor %}
    </div>
    <a class="carousel-control-prev" data-slide="prev"
href="#carouselExampleControls" role="button">
        <span aria-hidden="true" class="carousel-control-
prev-icon"></span>
        <span class="sr-only">Previous</span>
    </a>

```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		30


```

        <a class="carousel-control-next" data-slide="next"
href="#carouselExampleControls" role="button">
            <span aria-hidden="true" class="carousel-control-
next-icon"></span>
            <span class="sr-only">Next</span>
        </a>
    </div>

```

- Секція для ознайомлення з послугами які надаються в реабілітаційному центрі. На головній сторінці вони відображаються у вигляді фотографій для того щоб прочитати користувачу потрібно натиснути кнопку «Читати більше» перейшовши на сторінку відвідувач зможе прочитати необхідну йому інформацію:

```

    <div class="container mt-5">
        <div class="row">
            <div class="col-3">
                <h6 style="font-
size:25px;!important">Процедури</h6>
            </div>
        </div>
        <div class="row mt-4">
            {% for rehabil in reabils %}
            <div class="col-md-4 " >
                
            </div>
            {% endfor %}
        </div>
        <div class="row mt-3 d-flex justify-content-end ">
            <div >
                <a href="{% url 'rehabil' %}" class="btn
btn-outline-primary mr-2">Читати білше </a>
            </div>
        </div>
    </div>

```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		31

- Інформаційний блок «Про нас» який розповідає про організацію її діяльність:

```
<div class="container mt-2">
  <div class="row mt-3">
    <div class="col-5">
      <h6 style="font-size:25px;">Про нас</h6>
    </div>
  </div>

  <div class="row mt-4 d-flex justify-content-center">
    <div class="col-md-10">

      {% for information in informations %}
        
      {% endfor%}
    </div>
  </div>

  <div class="row d-flex justify-content-center">
    <div class="col-md-10 text-align-left mt-3" style="font-size:18px;">
      {% for information in informations %}
        {{information.text}}
      {% endfor%}
    </div>
  </div>
</div>
```

- Footer сайту який містить контактну інформацію про реабілітаційний центр і посилання на соціальні мережі:

```
<footer class="container mt-5 mb-4 pt-5 sticky-bottom">
  <div class="row">
    <div class="col-6 contact"><h6 >Контакти</h6></div>
  </div>
```

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		32

<i>Ми відкриті для пропозицій та ідей щодо розвитку Центру в напрямку психологічного розвантаження.</i>

```
<div class="row d-flex">
```

```
  <div class="col-xl-10 d-flex flex-column justify-content-between mt-3 ">
```

```
    {% for contact in contacts %}
```

```
    <a href="mailto:{{contact.email}}" class="text-dark font-weight-bold text-decoration-none">{{contact.email}}</a>
```

```
    {% endfor %}
```

```
    {% for contact in contacts%}
```

```
    <a href="tel:{{contact.phone}}" class="text-dark font-weight-bold text-decoration-none">{{contact.phone}}</a>
```

```
    {% endfor %}
```

```
</div>
```

```
  <div class="col-sm-2 col-sm mt-4 d-flex justify-content-around social " >
```

```
    <a href="#" >
```

```
      <i class="fab fa-facebook-f text-dark"></i>
```

```
    </a>
```

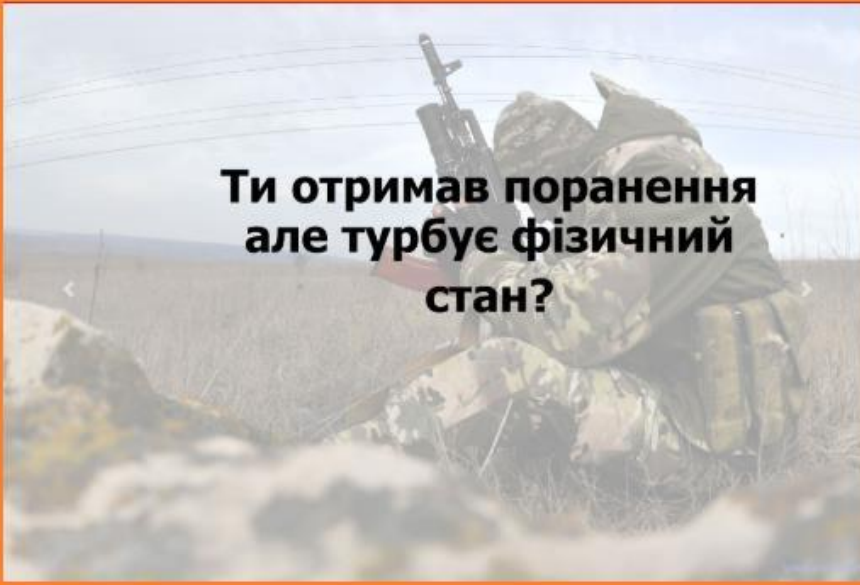
```
</div>
```

```
</div>
```

```
</footer>
```




					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		33

Нові крила
Зворотній зв'язок
Спонсорам




Ти отримав поранення але турбує фізичний стан?

Процедури

[Читати більше](#)

Про нас



Запрошуємо до нашого реабілітаційного релакс-центру Нові крила для учасників АТО/ОСБ та членів їх сімей. Робота в Центрі побудована за принципом «Рівний-Рівному», керівник і працівники мають досвід бойових дій в Донбасі. Послуги надають дипломовані спеціалісти. Активний простір з легкістю трансформується в затишний зал для проведення неформальних тематичних зустрічей, різноманітних майстер класів, групового терапевтичного спілкування, колективного перегляду фільмів. За побажанням відвідувачів.

Контакти

Ми відкриті для пропозицій та ідей щодо розвитку Центру в напрямку психологічного розвантаження.

ua.if.andriy20@gmail.com
+380000

f

Рисунок 3.1 – Поділ говоної сторінки на секції

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		34

3.2 Розробка сторінки зворотного зв'язку

Комунікація з клієнтом відграє дуже важливу роль для реабілітаційної організації. За допомогою форм зворотного зв'язку адміністратор сайту може дізнатися потрібну йому інформацію. Це дивно, але веб-сайти як і люди можуть бути комунікабельні або скромні і тихі. Останнім таланить менше, оскільки через свою закритість довіра оточуючих до даного ресурсу є набагато меншою. Для власника сайту головне - навчити «говорити» його ще змалку, тобто вчити проект комунікувати з відвідувачами. Для цього на сайті повинна бути форма зворотного зв'язку.

Наявність в інтерфейсі сайту розділу, де користувач може висловити свою думку або задати питання, є обов'язковим для будь-якого сучасного ресурсу. Це важливо, насамперед, для самого сайту. Для відвідувача важливо бути почутим, а не виступати в ролі німого глядача. Можливість залишити повідомлення робить його учасником в житті сайту. Завдяки реалізації самої ідеї спілкування всі соціальні мережі користуються високою популярністю серед користувачів. В моїй роботі я розробив такі форми (рис. 3.2):

- Прізвище Ім'я По-батькові;
- Рік народження ;
- Місце проживання;
- Номер телефону
- Проблема що торує;
- Перенесені травми або операції;
- Наявні хронічні захворювання;
- Необхідний вид реабілітації;
- Наявність пільг.

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		35

Прізвище Ім'я По батькові:

Ваш вік:

Місце проживання:

Проблема що торбує:

Перенесині травми, операції а також наяві захворювання :

Вид реабілітації:

Recent treatment:

Наявність пільг:

Відправити

Рисунок 3.2 – Форми зворотного зв'язку

Після заповнення всіх полів веб-браузер відправляє запит на сервер використовуючи метод відправки post що відправляє данні на сервер за допомогою протоколу НТТР. Після цього данні записуються в базу даних і публікуються в адміністративній панелі як окрема комірка (рис. 3.3).

Django адміністрування

Домівка > New_Wings > Supports

Виберіть suport щоб змінити

Дія: 0 з 1 обрано

SUPPORT

Підлісецький Андрій Васильович

1 suport

Рисунок 3.3 – Відображення даних клієнта

Робочий код представлено нижче:

```
<div class="container mt-5" id="m2">
  <div class="row">
    <div class="col-12">
      <h3>Зворотній зв'язок</h3>
      <p class="lead">
        Ви завжди можете написати нам про стан вашого здоров'я</p>
      <form action="">
        <div class="form-check">
          <div class="col-lg-5">
            <label for="inputName" class=" col-form-label">П і Б</label>
            <input type="text" class="form-control" id="inputFName" placeholder="First name">
          </div>
          <div class="col-lg-5 pt-3">
            <label for="inputAge" class=" col-form-label">Вік</label>
            <input type="number" class="form-control" id="inputAge" placeholder="Вік">
          </div>
          <div class="col-lg-5 pt-3">
            <label for="inputCity" class=" col-form-label">Місце проживання</label>
            <input type="text" class="form-control" id="inputCity" placeholder="Місце проживання">
          </div>
          <div class="col-lg-5 pt-3">
            <label for="inputEmail" class=" col-form-label">Email</label>
            <input type="email" class="form-control" id="inputEmail" placeholder="Email">
          </div>
        </div>
      </form>
    </div>
  </div>
</div>
```

					ДР.Іс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		37

```

<div class="col-lg-5 pt-3">
  <label for="inputMobail" class=" col-form-
label">Мобільний телефон</label>
  <input type="tel" class="form-control" id="inputMobail"
placeholder="Mobile">
</div>
<div class="col-lg-5 pt-3">
  <label for="selectRType" class=" col-form-label">Виберіть
вид реабілітації</label>
  <select class="form-control">
    <option>Психологічна</option>
    <option>Фізична</option>
  </select>
</div>
<div class="col-lg-5 pt-3">
  <label for="textareaInfo" class=" col-form-label">Опишіть
вашу поточну проблему , а також травми і операції, наявні хронічні
захворювання </label>
  <textarea class="form-control"
id="textareaInfo"></textarea>
</div>
<div class="col-lg-5 pt-3">
  <button type="submit" class="btn btn-
primary">Submit</button>
</div>
</form>
</div>
</div>
</div>

```

Таким чином було реалізовано одне з основних завдань проекту. А саме спрощення зв'язку між реабілітологом і його пацієнтом. За допомогою данії форми пацієнти зможуть розповісти про свої недуги визначитися з типом реабілітації, а також повідомити свої контактні дані, що набагато полегшує роботу як працівникам центру так і полегшує реєстрацію пацієнта на процедуру.

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		38

3.3 адміністративна панель і її налаштування

Адміністративна панель є найважливішою деталлю веб-сайту в різних випадках вона надає власнику змінювати контент без втручання робочий код проекту.

Таким чином власник веб ресурсу без лишніх клопотів може додати опис організації, змінити фотографію і інформацію про надання послуг а також додавати нові слайди в слайдер. Django адміністрування можна віднести до CMS (Системи керування вмістом) - це система для управління вмістом .Набір скриптів для створення, редагування і управлінням контентом сайту. До прикладу існують такі CMS:

- Wordpress призначений для блогів та інформаційних сайтів. WordPress має спрощеним інтерфейсом, орієнтованим на непрофесійного користувача;

- Joomla - система управління сайтом, безкоштовно розповсюджується в мережі Інтернет. При завантаженні CMS надає мінімальний набір інструментів, які згодом доповнює сам користувач у міру необхідності. Завдяки цьому знижується навантаження на сервер;

- OpenCart - популярна CMS, призначена для управління контентом інтернет-магазинів. Ця система поширюється безкоштовно, завантажити її можна з офіційного сайту. CMS має інтерфейс, зрозумілий і для власника сайту, і для його відвідувачів;

Django адміністрування використовує моделі як і створив в файлі models.py як комірки даних. Але для того, щоб моделі відтворювалися в адмін панелі, потрібно їх зареєструвати в файлі admin.py:

```
from django.contrib import admin
from .models import *
class InformationAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 1:
```

```

        return False
    else:
        return True

class SliderAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 5:
            return False
        else:
            return True

class SocialAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 3:
            return False
        else:
            return True

class ContactAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 1:
            return False
        else:
            return True

admin.site.register(Slider,SliderAdmin)
admin.site.register(Information,InformationAdmin)
admin.site.register(Contact,ContactAdmin)
admin.site.register(Social,SocialAdmin)
admin.site.register(Reabil)
admin.site.register(Sponsor)
admin.site.register(Donate)
admin.site.register(New)
admin.site.register(Suport)

```

					ДР.ІІс – 11.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		40

Там де:

- `from django.contrib import admin` - я імпортую модуль `admin`, тим самим завантажую метод рїстрації моделей;
- `from .models import *` з файлу `models.py` я імпортую всі моделі (п можливості можна імпортувати їх по одній вказавши назву після слова `import`);
- `class InformationAdmin (admin.ModelAdmin)` тут взявши назвву моделі і додваши до неї слово «Admin» я вказую що це вже не просто модель для бази даних , це модель адміністративної панелі . В даноий клас обмежує кількість екзимплярів моделей наприклад якщо кількість в моделі `Informations` є як один екзимпляр то кнопка «додати» що зазвичай знаходиться навпроти кожної моделі зникає (рис.3.4).

Адміністрування New_Wings

NEW_WINGS		
Contacts		Змінити
Donates	+ Додати	Змінити
Informations		Змінити
News	+ Додати	Змінити
Reabils	+ Додати	Змінити
Sliders	+ Додати	Змінити
Socials	+ Додати	Змінити
Sponsors	+ Додати	Змінити
Suports	+ Додати	Змінити

Рисунок 3.4 – Приклад видалення кнопки «Додати»

`admin.site.register(Slider, SliderAdmin)` - даний рядок коду реєструє модель в адміністративній панелі. В дужках вказується яка саме модель буде зареєстрована.

Розглянемо, як працює адміністративна панель. Спочатку в адреному вікні адміністратор повинен додати до адреси веб-сайту `/admin`. Так браузер розуміє, що користувачу потрібно потрапити імено в адмін. панель Django. Коли користувач перейде по заданій адресі передним відкриється вікно авторизації, в якому потрібно вести «Логін» і «Пароль» (рис.3.5).

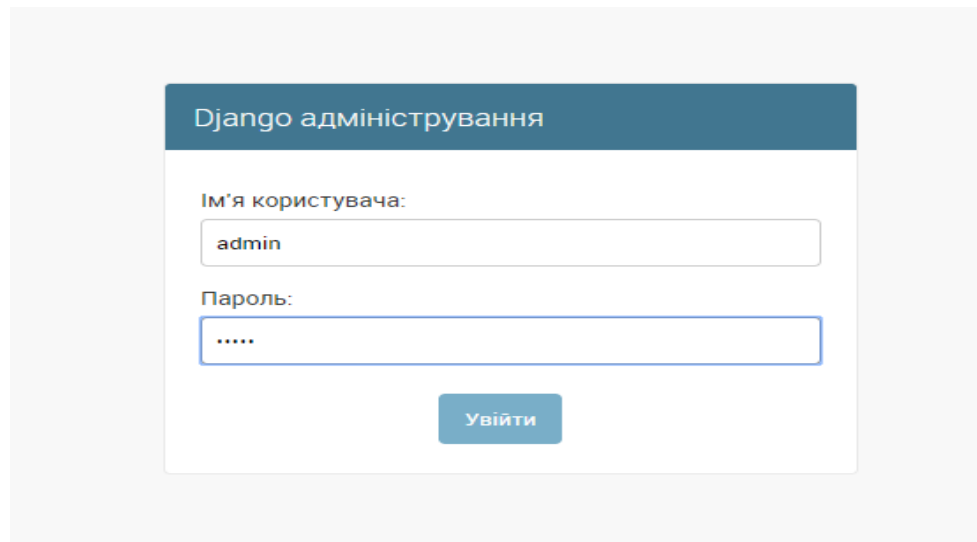


Рисунок 3.5 - Авторизація в адміністративну панель Django

Після успішної авторизації користувач побачить список моделей які може сміливо редагувати для цього призначені кнопки «Додати» (позначена зеленим плюсом) і кнопка змінити (жовтим мінусом) також можна побачити свою недавню активність що знаходиться з права від списку моделей (рис.3.6).

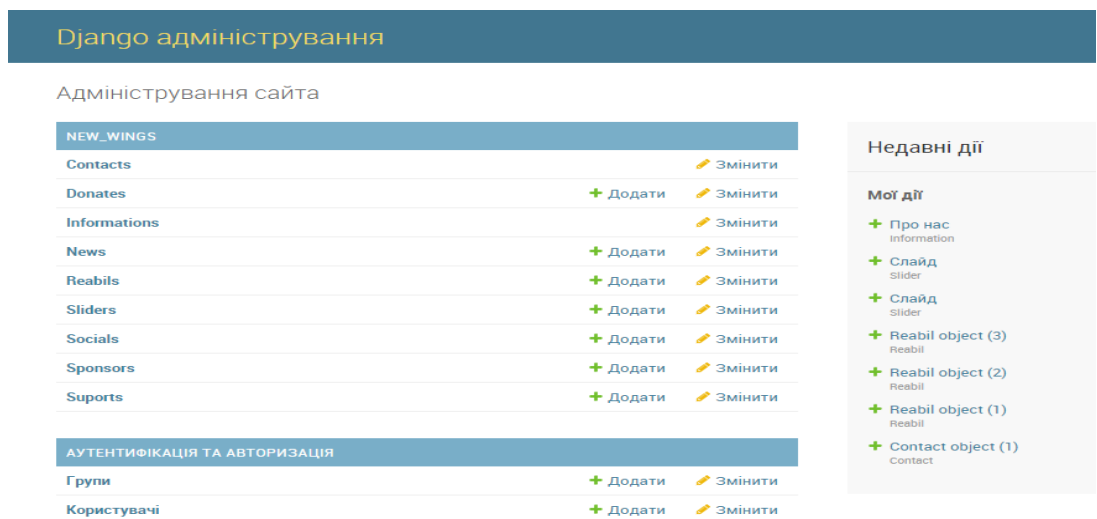


Рисунок 3.6.- Інтерфейс Django адміністрування

Також адміністратор може видаляти контент. Для цього варто вибрати модель і навпроти потрібного поля поставити галочку, потім в рядку де пише «Дія:» вибрати видалити обранні (рис 3.7).

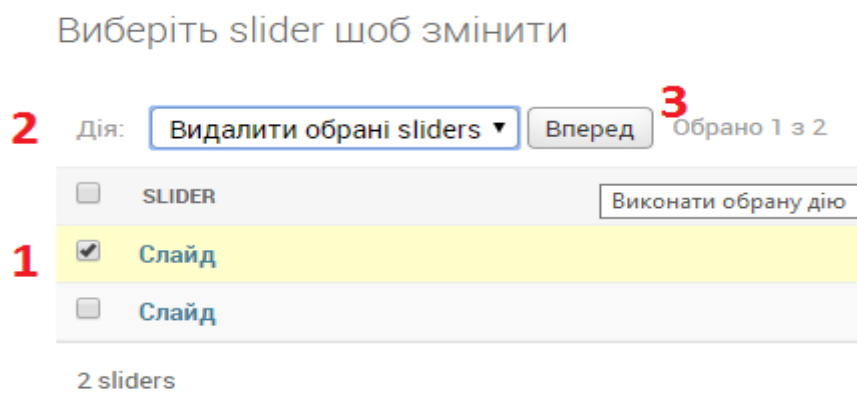


Рисунок 3.7 – Видалення об’єкту

Для того, щоб додана інформація появилася на сторінці використовується шаблонізатор Jinja, за допомогою якого можна працювати з шаблонами (html-сторінками). Для початку в файлі view.py потрібно написати функцію яка б визивала нашу модель і передавала весь її зміст як об’єкти. Приклад коду даної функції можна побачити нижче:

```
def home(request):  
    return render(request, 'new_wings/index.html', {  
        'sliders': slider.objects.all(),  
        'informations': information.objects.all(),  
        'contacts': contact.objects.all(),  
        'socials': social.objects.all(),  
        'reabils': reabil.objects.all(),  
    })
```

Спочатку я оголосив функцію та дав її назву «home». Вона повертає html сторінку. А потім вже, як додаткові аргументи, я задав додавання об’єктів. Візьмемо до прикладу модель «slider». Для того щоб отримати весь її зміст потрібно вказати 'sliders': slider

Але це не все наступним кроком потрібно додати до нашої розмітки слайдера `{% for slider in slides %}` - таким чином цикл `for` проходить по всіх об'єктах, що є в даній моделі. Далі в тезі `` я використав наступний код `{{slider.screen.url}}`, він вказує що з моделі `slider` тут потрібно вставити поле `screen`. Таким чином, на сторінці успішно появився слайдер (рис.3.8).

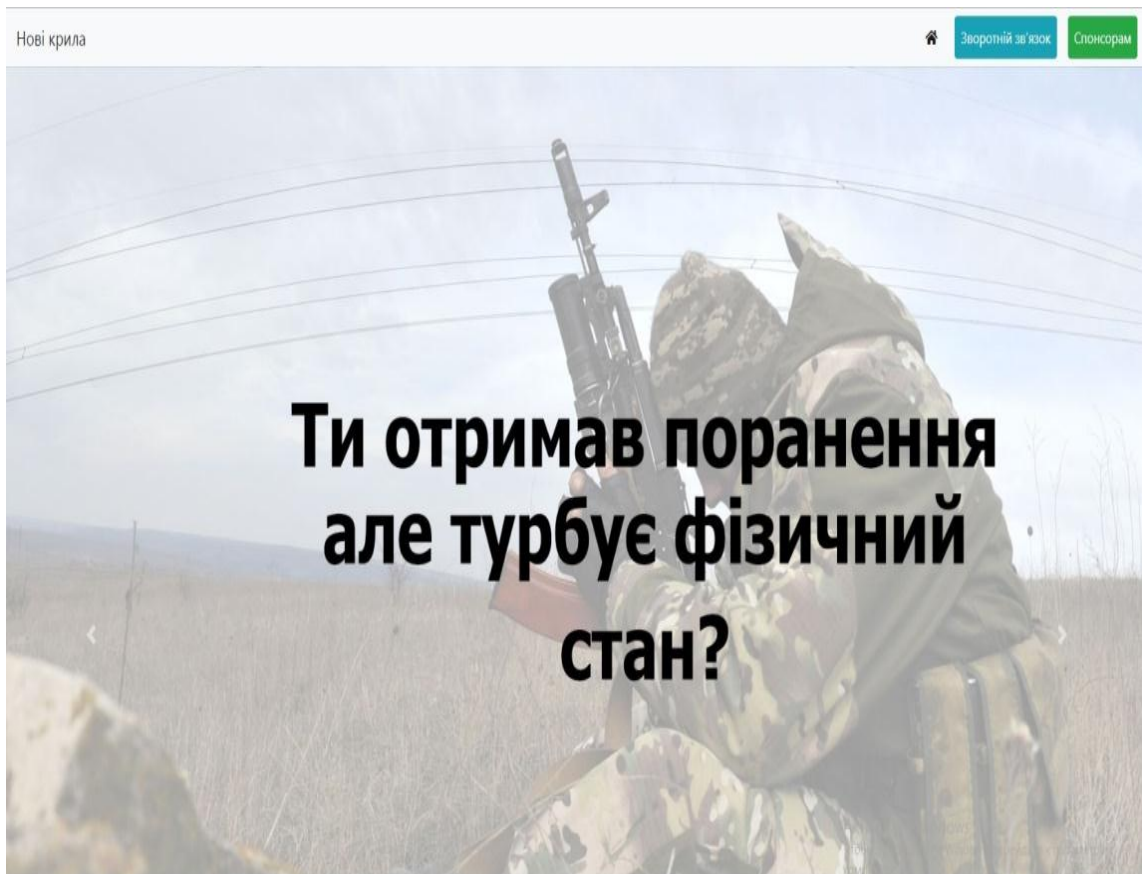


Рисунок 3.8 – Доданий слайдер

Отже, в результаті мною було розроблено: зручну адміністративну панель. звдяки якій персонал закладу може не вмішуючись в програмний код швидко змінити контент (фото, відео, текст). В адміністративній панелі міститься пункт, де аміністратор зможе дізнатися данні про пацієнтів та вид реабілітації, яка потрібна конкретному пацієнту.

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		44

ВИСНОВКИ

Під час виконання дипломного проекту було розроблено веб-ресурс для реабілітаційного центру учасників АТО та членів їх сімей «Нові Крила».

Під час роботи було досліджено процесу формування веб-представництва у мережі Інтернет для інформування учасників АТО про надання медичних послуг центром реабілітації, виявлення недуг і проблем психологічного або фізичного характеру, та створення інформаційного веб-ресурсу для спрощення зв'язків між центром і пацієнтами та центром і спонсорами.

Веб-ресурс реалізований за допомогою таких веб технологій як - Bootstrap, Django, Jade (HTML препроцесор), Scss (CSS препроцесор). Проект розроблений з розрахунком на подальше удосконалення за бажанням замовника.

Результатом роботи є повністю функціональний, протестований і готовий до використання веб-ресурс.

В дипломну роботу включено весь матеріал який описує розробку і функціонування проекту.

Дипломний проект виконано у повній відповідності до завдання і всіма нормативними вимогами. В додатку міститься робочий код програми.

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		45

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Про доступ до публічної інформації: Закон України від 13.01.2011 р. №2939-VI. URL: <https://zakon.rada.gov.ua/laws/show/2939-17>
2. Про інформацію : Закон України від 02.10.1992 р. № 2657 XII. URL: <https://zakon.rada.gov.ua/laws/show/2657-12> (дата зверення: 15.12. 2018).
3. Про авторське право і суміжні права : Закон України від 23 грудня 1993 р. № 3792-XII URL: <https://zakon.rada.gov.ua/laws/show/3792-12>
4. Revived Soldiers Ukraine : волонтерська організація.
URL: www.rsukraine.org.
5. Svelte : веб фреймворк. URL: <https://svelte.dev/>.
6. Stencil : веб фреймворк. URL: <https://stenciljs.com/>
7. Cocoa.: веб фреймворк.
URL: <https://developer.apple.com/library/archive/documentation/General/Conceptual/DevPedia-CocoaCore/Cocoa.html>
8. jqueryui.: веб фреймворк. URL: <https://jqueryui.com/>
9. Angular веб фреймворк URL <https://angularjs.org/>
10. Redux веб фреймворк URL <https://redux.js.org/>
11. Vuejs веб фреймворк URL <https://vuejs.org/>
12. Ruby on Rails веб фреймворк URL <https://rubyonrails.org/>.
13. Стаття з Вікіпедії : Модель-вид-контролер
URL: <https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5>
14. Веб креатор. Навчальни веб сайт.
URL: <https://web-creator.ru/articles/mvc>
15. Adobe Photoshop. Веб редактор.
URL: <https://www.adobe.com/ua/products/photoshop.html>
16. Стаття з вікіпедії. MySQL.
URL: <https://uk.wikipedia.org/wiki/MySQL>

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		46

17. Чаков С.,Штрауб Б. Github для професыйного програміста – СПб.:Питер,2016. - 496 с.
18. Рейтц К., Шлюсер Т. Автостопом по Python. – СПб.:Gbnth 2017. – 336с
19. <https://python-scripts.com/virtualenv>
20. Зубов В.С. Справочник программиста. Базовые методы решения графовых задач и сортировки.- М.: "Филинь", 1999.- 256 с.
21. Оре О. Теория графов.- М.: 1980.
22. Ю.Н.Тюрин, А.А.Макаров. Анализ данных на компьютере.- М.: ИНФРА-М, Финансы и стат., 1995.- 384 с.
23. В.Н.Нефедов, В.А.Осипова. Курс дискретной математики: Учеб. пос.- М.: Изд-во МАИ, 1992.- 264 с.
24. Форта Б. SQL за 10 минут. Москва: «Вильямс», 2014 – 340 с.
25. Хоган Б. «HTML5 и CSS3. Веб-по стандартам нового поколения». Санкт-Петербург: Питер, 2011. – 272 с.
26. Microsoft. Центр загрузки. Microsoft Visual Studio Learning Pack 2.0 [Электронный ресурс] – Режим доступа: <http://www.microsoft.com/downloads/details.aspx?displaylang=ru&FamilyID=0ce3cbbd-7fc7-410b-8c2c-e18d1c60a6cd>
27. Дэвид Флэнаган, JavaScript – Подробное руководство 6-ое издание Пер. с англ. — М.: Издательский дом «Вильямс», 2015. — 300 с.
28. Морето С. JavaScript и jQuery. Исчерпывающее руководство. / С. Морето — Пер. с англ. Издательский дом «Вильямс», 2017 — 880 с.
29. Морето С. Bootstrap в примерах. / С. Морето — Пер. с англ. М.: Издательский дом «ДМК Пресс», 2017. — 314 с.
30. Дэвид Флэнаган, JavaScript. Карманный справочник. Пер. з англ. – «Питер», 2015. – 600с.

					ДР.Пс – 11.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		47

Додаток А

Файл models.py

```
from django.db import models
from django.core.exceptions import ValidationError
from django.core.validators import MaxValueValidator,
MinValueValidator

class Slider(models.Model):
    name = models.CharField(blank=True, max_length=50)
    screen = models.ImageField(blank=True, null=True,
upload_to='image')

    def __str__(self):
        return 'Слайд'.format(self.id)

class Information(models.Model):
    image = models.ImageField(blank=True, null=True,
upload_to='image')
    text = models.CharField(max_length=500)

    def __str__(self):
        return 'Про нас'.format(self.id)

class Contact(models.Model):
    phone = models.CharField(max_length=150)
    email = models.EmailField(max_length=254)

class Social(models.Model):
    link = models.URLField(max_length=200)
```

```
        icon = models.ImageField(blank=False, null=True,
upload_to='image')
```

```
class Reabil(models.Model):
    text = models.TextField(max_length=1000)
    image = models.ImageField(blank=True, null=True,
upload_to='image')
```

```
class Donate(models.Model):
    account = models.CharField(max_length=100)
```

```
class Sponsor(models.Model):
    logo = models.ImageField(blank=True, null=True,
upload_to='image')
```

```
class New(models.Model):
    title = models.CharField("Заголовок", max_length=100)
    images = models.ImageField(blank=False, null=True,
upload_to='image/')
    text = models.TextField("Опис")
    created = models.DateTimeField("Дата створення",
auto_now_add=True)
```

```
        keywords = models.CharField("Ключові слова",
max_length=50)
```

```
    def __str__(self):
        return self.title
```

```
class Suport(models.Model):
    full_name = models.CharField("Прізвище Ім'я По
батькові", max_length=150)
```

```

age = models.IntegerField("ваш вік", default=1,

validators=[MaxValueValidator(10), MinValueValidator(1)])
    city = models.TextField("Місце проживання",
max_length=200)
        problem = models.TextField("Проблема що торбує")
        live_through = models.TextField("Перенесині
травми, операції а також наяві захворювання: ", max_length=200)
        recent_treatment = models.CharField(max_length=200)
        rehabil_type = models.CharField("Вид реабілітації",
max_length=100, )
        exemption = models.BooleanField("Наявність пільг", )
    def __str__(self):
        return self.full_name

```

Файл view.py

```

from django.shortcuts import render ,HttpResponseRedirect
from django.core.mail import send_mail,BadHeaderError
from .forms import *
from .models import*
def home(request):
    return render(request, 'new_wings/index.html', {
        'sliders': Slider.objects.all(),
        'informations':Information.objects.all(),
        'contacts':Contact.objects.all(),
        'socials':Social.objects.all(),
        'reabils': Reabil.objects.all(),
    })

```

```

def support(request):
    form = SuportForm(request.POST or None)
    if request.method == "POST":
        print(request.POST)
        if form.is_valid():
            form = form.save()
            return HttpResponseRedirect(request.path)
    return render(request, 'new_wings/suport.html',
locals(),{
        'contacts': Contact.objects.all(),
        'socials': Social.objects.all()
    })
def rehabil(request):
    return render(request, 'new_wings/rehabilitation.html', {
        'contacts': Contact.objects.all(),
        'socials': Social.objects.all(),
        'reabils': Reabil.objects.all(),
    })
def donate(request):
    return render(request, 'new_wings/sponsors.html', {
        'contacts': Contact.objects.all(),
        'socials': Social.objects.all(),
        'donates': Donate.objects.all(),
        'sponsors': Sponsor.objects.all()

    })
def news(request):
    return render(request, 'new_wings/news.html', {
        'news': New.objects.all()
    })

```

Файл admin.py

```
from django.contrib import admin
from django.contrib import admin
from .models import *
class InformationAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 1:
            return False
        else:
            return True
class SliderAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 5:
            return False
        else:
            return True
class SocialAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 3:
            return False
        else:
            return True
class ContactAdmin (admin.ModelAdmin):
    def has_add_permission(self, request):
        num_objects = self.model.objects.count()
        if num_objects >= 1:
            return False
        else:
            return True
```

```
admin.site.register(Slider,SliderAdmin)
admin.site.register(Information,InformationAdmin)
admin.site.register(Contact,ContactAdmin)
admin.site.register(Social,SocialAdmin)
admin.site.register(Reabil)
admin.site.register(Sponsor)
admin.site.register(Donate)
admin.site.register(New)
admin.site.register(Suport)
# admin.site.register(MetaTag)
#

# admin.site.register(FirstSection)
# admin.site.register(ServisList)
# admin.site.register(SecondSection)
# admin.site.register(FourthSection)
# admin.site.register(ContactSectiion)
```

Додаток Б

Файл index.html

```
<!DOCTYPE html>
<html lang="uk">

<head>
  <meta charset="UTF-8">
  <meta content="width=device-width, initial-scale=1.0"
name="viewport">

  <meta content=""
      name="description"/>
  <meta content="Нові Крила" name="keywords"/>
  <title>Нові Крила</title>
  <link crossorigin="anonymous"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/boo
tstrap.min.css"
      integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T
" rel="stylesheet">

  <link crossorigin="anonymous"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
      integrity="sha384-
50oBUHEmvpQ+1lW4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJK28anvf
" rel="stylesheet">

  <link href="/images/favicon.ico" rel="shortcut icon"
type="image/x-icon">
```



```

    <link
href="https://fonts.googleapis.com/css?family=Roboto:300,700&
;subset=cyrillic,cyrillic-ext"
    rel="stylesheet">
    <link href="https://fonts.googleapis.com/css?family=Anton"
rel="stylesheet">
    <link rel="stylesheet" type="text/css" media="screen"
href="{% static 'new_wings/style.css' %}"/>
</head>

<body>

{% include "new_wings/header.html"%}

<div class="carousel slide" data-ride="carousel"
id="carouselExampleControls">
    <div class="carousel-inner">
        {% for slider in sliders %}
            <div class="carousel-item {% if forloop.first %}
active{% endif %}">
                
            </div>
        {% endfor %}

    </div>
    <a class="carousel-control-prev" data-slide="prev"
href="#carouselExampleControls" role="button">
        <span aria-hidden="true" class="carousel-control-prev-
icon"></span>
        <span class="sr-only">Previous</span>
    </a>
    <a class="carousel-control-next" data-slide="next"
href="#carouselExampleControls" role="button">
        <span aria-hidden="true" class="carousel-control-next-
icon"></span>

```

```

        <span class="sr-only">Next</span>
    </a>
</div>
<div class="container mt-5">
    <div class="row">
        <div class="col-3">
            <h6 style="font-size:25px;!important">Процедури</h6>
        </div>
    </div>
    <div class="row mt-4">
        {% for rehabil in reabils %}
        <div class="col-md-4 " >

        </div>
        {% endfor %}

    </div>
    <div class="row mt-3 d-flex justify-content-end ">
        <div >
            <a href="{% url 'rehabil' %}" class="btn btn-outline-primary mr-2">Читати білше </a>
        </div>

    </div>
</div class="container mt-4">
    <div class="row">
    </div>
</div>

```

```

</div>
<div class="container mt-2">
  <div class="row mt-3">
    <div class="col-5"><h6 style="font-size:25px;">Προ
    hac</h6></div>
  </div>

  <div class="row mt-4 d-flex justify-content-center">
    <div class="col-md-10">

      {% for information in informations %}
        
      {% endfor%}
    </div>

  </div>
  <div class="row d-flex justify-content-center">
    <div class="col-md-10 text-align-left mt-3"
style="font-size:18px;">
      {% for information in informations %}
        {{information.text}}
      {% endfor%}
    </div>
  </div>
</div>

{% include "new_wings/footer.html" %}
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.m
in.js"></script>
<script crossorigin="anonymous"

```

```

        integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo
"
        src="https://code.jquery.com/jquery-
3.3.1.slim.min.js"></script>
<script crossorigin="anonymous"
        integrity="sha384-
U02eT0CpHqdSjQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1
"

src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd
/popper.min.js"></script>
<script crossorigin="anonymous"
        integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDs4x0xIM+B07jRM
"

src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootst
rap.min.js"></script>
</body>

</html>

```

Файл suport.html

```

{% load static %}
<!DOCTYPE html>
<html lang="en">
<link
href="https://fonts.googleapis.com/css?family=Noto+Serif&display
=swap" rel="stylesheet">
<head>
    <meta charset="UTF-8">
    <meta content="width=device-width, initial-scale=1.0"
name="viewport">
    <title>Зворотній зв'язок</title>
    <link crossorigin="anonymous"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
        integrity="sha384-
50oBUHEmvpQ+1lW4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJK28anvf
" rel="stylesheet">
    <link crossorigin="anonymous"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/boo

```

```

tstrap.min.css"
    integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T
" rel="stylesheet">

<link
href="https://fonts.googleapis.com/css?family=Roboto:300,700&
;subset=cyrillic,cyrillic-ext"
    rel="stylesheet">
    <link href="https://fonts.googleapis.com/css?family=Anton"
rel="stylesheet">
    <link href="style.css" rel="stylesheet">
</head>

<body>
{% include "new_wings/header.html"%}
<div class="container">
    <div class="row">
        <div class="col-10">
            <form method="post">{% csrf_token %}

                {{ form }}

        <div class="row mt-2 ">
            <div class="col-md-2 mr-auto"><button class="btn btn-
primary" type="submit">Відправити</button></div>
        </div>
    </form>
    </div>

</div>

    {% include "new_wings/footer.html" %}
    <script crossorigin="anonymous"
    integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo
"
    src="https://code.jquery.com/jquery-
3.3.1.slim.min.js"></script>
    <script crossorigin="anonymous"
    integrity="sha384-
U02eT0CpHqdsJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1
"
    src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd
/popper.min.js"></script>
    <script crossorigin="anonymous"
    integrity="sha384-

```

```
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM
"
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootst
rap.min.js"></script>
</body>
```

sponsors.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta content="width=device-width, initial-scale=1.0"
name="viewport">
  <title>Спонсорам</title>
  <link crossorigin="anonymous"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/boo
tstrap.min.css"
  integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T
" rel="stylesheet">
  <link crossorigin="anonymous"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
  integrity="sha384-
50oBUHEmvpQ+1lW4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJK28anvf
" rel="stylesheet">
  <link href="/images/favicon.ico" rel="shortcut icon"
type="image/x-icon">
  <link
href="https://fonts.googleapis.com/css?family=Roboto:300,700&
;subset=cyrillic,cyrillic-ext"
```

```

        rel="stylesheet">
        <link href="https://fonts.googleapis.com/css?family=Anton"
rel="stylesheet">
        <link rel="stylesheet" type="text/css" media="screen"
href="{% static 'new_wings/style.css' %}"/>
</head>
<body>
    {% include 'new_wings/header.html' %}
    <div class="container mt-5">
        <div class="row">
            <div class="col-4">
                <h1>Наші спонсори</h1>
            </div>
        </div>
        <div class="row mt-4">
            {% for sponsor in sponsors %}
            <div class="col-mb-1 ">
                
            </div>
            {% endfor %}

        </div>
    </div>
    <div class="container mt-5">
        <div class="row d-flex justify-content-center">
            <div class="col-11">
                <hr class="bg-secondary">
            </div>
        </div>
        <div class="row">
            <div class="col-8">

```

Центр створено волонтерськими зусиллями на кошти спонсорів та меценатів. Для учасників бойових дій послуги надаються на безоплатній основі, благодійні внески для подальшої

ефективної роботи Центру вітаються.

```

</div>
<div class="w-100"><hr></div>

<div class="col-8">
    Наші реквізити:
    <ul>
        {%for donate in donates %}
        <li>{{donate.account}}</li>
        {% endfor %}
    </ul>

</div>

</div>

</div>
{% include 'new_wings/footer.html' %}
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.m
in.js"></script>
<script crossorigin="anonymous"
    integrity="sha384-
q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo
"
    src="https://code.jquery.com/jquery-
3.3.1.slim.min.js"></script>
<script crossorigin="anonymous"
    integrity="sha384-
U02eT0CpHqdsJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1
"
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd
/popper.min.js"></script>
<script crossorigin="anonymous"
    integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM

```



```
"
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootst
trap.min.js"></script>
</body>
</html>
```

Файл rehabilitation.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta content="width=device-width, initial-scale=1.0"
name="viewport">
  <title>Title</title>

  <link crossorigin="anonymous"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/boo
tstrap.min.css"
  integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T
" rel="stylesheet">

  <link crossorigin="anonymous"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
  integrity="sha384-
50oBUHEmvpQ+1lW4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJK28anvf
" rel="stylesheet">

  <link href="/images/favicon.ico" rel="shortcut icon"
type="image/x-icon">
  <link
href="https://fonts.googleapis.com/css?family=Roboto:300,700&amp;
```

```

;subset=cyrillic,cyrillic-ext"
    rel="stylesheet">
    <link href="https://fonts.googleapis.com/css?family=Anton"
rel="stylesheet">
    <link rel="stylesheet" type="text/css" media="screen"
href="{% static 'new_wings/style.css' %}"/>
</head>
<body>
    {% include "new_wings/header.html" %}
    <div class="container mt-5">
        {% for rehabil in reabils %}
        <div class="row mt-3">
            <div class="col-md-8" style="font-size:15px;">
                {{rehabil.text}}
            </div>
            <div class="col-md-4">
                
            </div>
        </div>
        {% endfor %}
    </div>
    {% include "new_wings/footer.html" %}
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.m
in.js"></script>
<script crossorigin="anonymous"
    integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo
"
    src="https://code.jquery.com/jquery-
3.3.1.slim.min.js"></script>
<script crossorigin="anonymous"
    integrity="sha384-
UO2eT0CpHqdSJK6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1
"

```

```

src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd
/popper.min.js"></script>
<script crossorigin="anonymous"
    integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDs4x0xIM+B07jRM
"
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootst
rap.min.js"></script>
</body>
</html>

```

Файл Header.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <nav class="navbar sticky-top navbar-expand-lg navbar-light
bg-light ">
        <button aria-controls="navbarText" aria-expanded="false"
aria-label="Toggle navigation" class="navbar-toggler border-0"
            data-target="#navbarText" data-toggle="collapse"
type="button">
            <span class="navbar-toggler-icon "></span>
        </button>
        <a class="navbar-brand" href="#">Нові крила</a>
        <div class="collapse navbar-collapse" id="navbarText">
            <ul class="navbar-nav ml-auto">
                <li class="nav-item active">
                    <a class="nav-link btn" href="{% url 'home'
%}"><i class="fas fa-home"></i><span class="sr-
only">(current)</span></a>
                </li>
                <li class="nav-item col-md">
                    <a class="nav-link text-center text-dark"

```

```
href="{% url 'support' %}">Зворотній зв'язок</a>
    </li>
    <li class="nav-item ml-3">
        <a class="nav-link btn btn-success text-white"
href="{% url 'donate' %}">Спонсорам</a>
    </li>
</ul>
<span class="navbar-text">
</span>
</div>
</nav>
</body>
</html>
```

БІБЛІОГРАФІЧНА ДОВІДКА

Тема дипломної роботи: *Розробка інформаційного веб-ресурсу реабілітаційного центру для воїнів АТО «Нові Крила»*

Обсяг пояснювальної записки: _____ 71 _____

Перелік графічних матеріалів:

- рисуноків _____ 7 _____
- додатків _____ 2 на 16 _____ аркушів.

Дата закінчення дипломного проекту: “__” _____ 2019 р.

Студент – дипломник _____
(підпис) (розшифровка підпису)