

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Тернопільський національний економічний університет**  
**Факультет комп'ютерних інформаційних технологій**  
Кафедра комп'ютерних наук

**ПЕТЕЛЬКО Василь Всеволодович**

**Web-сайт мережі спортивно-оздоровчих центрів  
"Фітнес"/ Website for network of sports and  
wellness centers "Fitness"**

напрямок підготовки: 6.050103 - Програмна інженерія  
фахове спрямування - Програмне забезпечення систем

Бакалаврська дипломна робота

Виконав студент групи ПЗС-41  
В. В. Петелько

---

Науковий керівник:  
к.т.н., старший викладач  
ВОЙТЮК І.Ф.

---

Бакалаврську дипломну роботу  
допущено до захисту:

"\_\_" \_\_\_\_\_ 20\_\_ р.

Завідувач кафедри  
\_\_\_\_\_ **А. В. Пукас**

## РЕЗЮМЕ

Дипломна робота містить 70 сторінок, таблиці, 45 рисунків, список використаних джерел та 4 додатки.

Метою дипломної роботи є розробка Web-сайту мережі спортивно-оздоровчих центрів "Fitness" для можливості надання інформації про послуги, можливості замовити послугу.

Об'єктом дослідження є процес замовлення послуги, надання інформації про послуги і центр.

Предметом дослідження є застосування сучасних інформаційних технологій створення програмного продукту для мережі спортивно-оздоровчих центрів «Fitness» з метою надання можливості клієнтам отримати інформацію про послуги та здійснити запис на неї, а персоналу – проводити віддалене адміністрування.

Методи розробки базуються на використанні веб-технологій.

Одержані результати полягають в розробці інтерактивного веб-сайту спортивно-оздоровчих центрів «Fitness».

## SUMMARY

Thesis contains 76 pages, tables, 47 figures, list of sources, 4 application. The aim of the thesis is develop web-application sports health centers, to be able to provide information on services.

Object of research is the process of be able to provide information about services, opportunity to book the service.

The subject of research is the use of modern information technologies to create software for sport-welness center "Fitness" to enable clients get information about services, and make recordind on it, and staff conduct remote administration.

Methods of developing based on the use of web technologies and relational database management system.

The results are developing an interactive web service "Fitness".

## ЗМІСТ

ВСТУП .....	9
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
1.1. Коротка характеристика об'єкту управління .....	11
1.2. Опис предметної області.....	12
1.3. Огляд та аналіз існуючих аналогів .....	21
1.4. Специфікація вимог до програмного продукту .....	29
Висновок до першого розділу .....	34
РОЗДІЛ 2. ПРОЕКТУВАННЯ .....	36
2.1. Розробка архітектури програмної системи .....	36
2.2. Проектування структури бази даних. ....	40
Висновок до другого розділу .....	41
РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ .....	42
3.1. Програмна реалізація проекту .....	42
3.2. Програмна реалізація бази даних .....	44
Висновок до третього розділу .....	44
РОЗДІЛ 4. ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ .....	45
4. Тестування.....	45
4.2 Розгортання програмного продукту .....	48
4.3 Інструкція користувача .....	49
Висновок до четвертого розділу.....	53
Список використаної літератури.....	54
Додаток А Глосарій.....	55
Додаток Б Специфікація вимог до програмного продукту .....	57
Додаток В Результат автоматизованого тестування.....	61
Додаток Е Лістинг коду функцій адміністратора .....	65
Додаток F Лістинг коду панелі адміністратора .....	68
Додаток G Лістинг коду зворотнього зв'язку .....	69
Додаток H лістинг коду процесу запису на послугу.....	70

## ВСТУП

Зараз люди намагаються слідувати за собою, підтримувати себе в хорошому стані. Тому відвідування спортивно-оздоровчого центру є невід'ємною частиною сьогодення.

### *Актуальність теми*

Велика кількість спортивних центрів мають власні веб-сайти, але там немає реалізованого функціоналу, який би давав можливість записатися на послугу онлайн. Зазвичай щоб записатися, потрібно йти в приймальню центру. Тому розробка веб-сервісу, який надаватиме можливість користувачу записуватися на послугу в режимі онлайн є актуальною. Такий програмний продукт пришвидшить та спростить процес отримання послуги в центрі.

На сьогодні будь-яка організація здійснює рекламу послуг, окремо від цього працює і система планування графіку надання цих послуг. Розроблений продукт дозволить оптимізувати час клієнта, а також працівників, які працюють у різних фітнес-центрах, що знаходяться у різних місцях, а всі необхідні функції – зібрані в одному додатку.

*Метою роботи* є дослідження та аналіз процесу надання клієнту отримання необхідної інформації та замовлення послуги, використовуючи графічний інтерфейс веб-сайту. Відштовхуючись від мети, роботу можна поділили на такі задачі:

1. Дослідити веб-сайти, які надають користувачам отримання інформації, запис на послугу. На основі проведеного аналізу розробити функціональні та нефункціональні вимоги до майбутнього програмного продукту.
2. Розробити архітектуру програмної системи та спроектувати структуру бази даних.
3. Зробити вибір мов програмування та технологій для програмної реалізації описаного продукту.
4. Провести тестування розробленого програмного продукту.

Методи, засоби та технології розробки

В процесі розробки використовуватимуться наступні мови програмування та технології:

- PHP – інтерпретована мова програмування для виконання скриптів та генерування HTML сторінок на стороні сервера.

- JavaScript – об'єктна мова програмування, що виконується на стороні клієнта в браузері.

- MySql – система управління реляційними базами даних.

- SQL – мова запитів для взаємодії з сервером баз даних.

#### *Практичне значення одержаних результатів*

Розроблений веб-сайт дозволить використовувати його в майбутньому для здійснення планування графіку тренувань за допомогою мережі Інтернет.

Об'єктом дослідження є процеси які дозволяють працювати з послугами, забезпечать перегляд інформації, зв'язок з адміністратором.

Предметом – методи які дозволяють реалізувати вище перераховані процеси.

## РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Коротка характеристика об'єкту управління

Напрямом діяльності досліджуваного об'єкту є надання своїх ресурсів клієнтам для задоволення їх потреб, а саме потреб у спорті. Адже не кожен може дозволити собі обладнати вдома зал для занять спортом. Відсутність досвіду і настанов тренера може привести до травматизму. Саме для вирішення даної проблеми було відкрито центр «Fitness».

Щоб підтримувати себе, своє тіло в хорошому стані люди відвідують спеціально обладнані, з відповідним персоналом центри. Це дозволяє людям слідкувати за станом свого здоров'я, покращити свої фізичні показники та стан організму в цілому. Заняття спортом дозволяють зняти фізичне і психологічне навантаження в людини, що позитивно позначається на її самопочутті.

Для задоволення вище переведених потреб люди відвідують спеціальні спортивно-оздоровчі центри. Спортивно-оздоровчий центр має мережу відділень і орієнтований на надання відповідних послуг своїм клієнтам. Мережа спортивно-оздоровчих центрів «Fitness» обладнана найсучаснішим спортивним обладнанням. У цій мережі працює висококваліфікований персонал.

Для того, щоб отримати інформацію, яка цікавить користувача, він повинен прийти до одного з центрів в приймальню адміністратора і дізнатися інформацію, яка його цікавить у адміністратора. Для автоматизації процесу необхідно розробити веб-систему.

Розроблювана система покликана надати наступну інформацію клієнту:

- про центр його розташування;
- робочий графік;
- послуги, які він надає;
- можливість записатися на послугу;
- написати листа адміністратору,

а адміністратору:

- додавати послуги;

- видаляти послуги;
- редагувати послуги.

Дана система дозволить користувачу отримати всю необхідну інформацію не виходячи з дому. Це дозволить суттєво зекономити час клієнта.

На рисунку 1.1 представлена схем організаційної структури центру.



Рис. 1.1 - організаційна структура центру

Фітнес центром керує директор. Йому підпорядковуються адміністратори відділень, адміністратору підпорядковується персонал який працює в відділенні(менеджери, бухгалтера, тренера і т.д.).

Послуги які надає центр:

- Тренажерний зал
  - Заняття з тренером
  - Кардіотренажери
  - Силові заняття
- Групові заняття:
  - Upperbody
  - TABS
  - Functional
  - Streching
  - Contemporary
  - Східні танці або по іншому танець живота.

- Rock-n-roll

- Та інше

- Масаж

### 1.2. Опис предметної області

Для належної роботи веб-сервісу мережі спортивно-оздоровчих центрів необхідне виконання таких бізнес процесів(рисунок 1.2):

- додавання послуги;
- редагування послуги;
- видалення послуги;
- перегляду інформації;
- авторизація адміністратора;
- відправка листа адміністратору;
- запис на послугу.

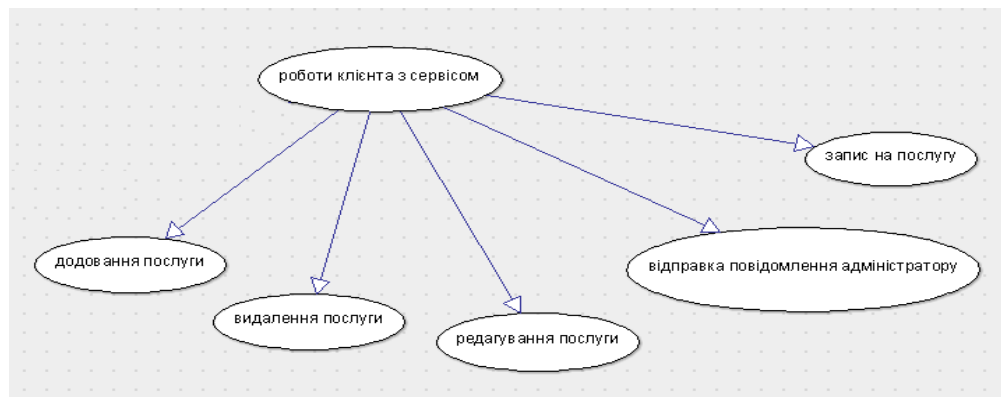


Рис. 1.2 . Діаграма бізнес-процесів розроблюваного програмного продукту



Розглянемо детальніше кожен з представлених на рисунку 1.2 бізнес-процесів.

На рисунку 1.3 бачимо діаграму функції процесу додавання послуги.

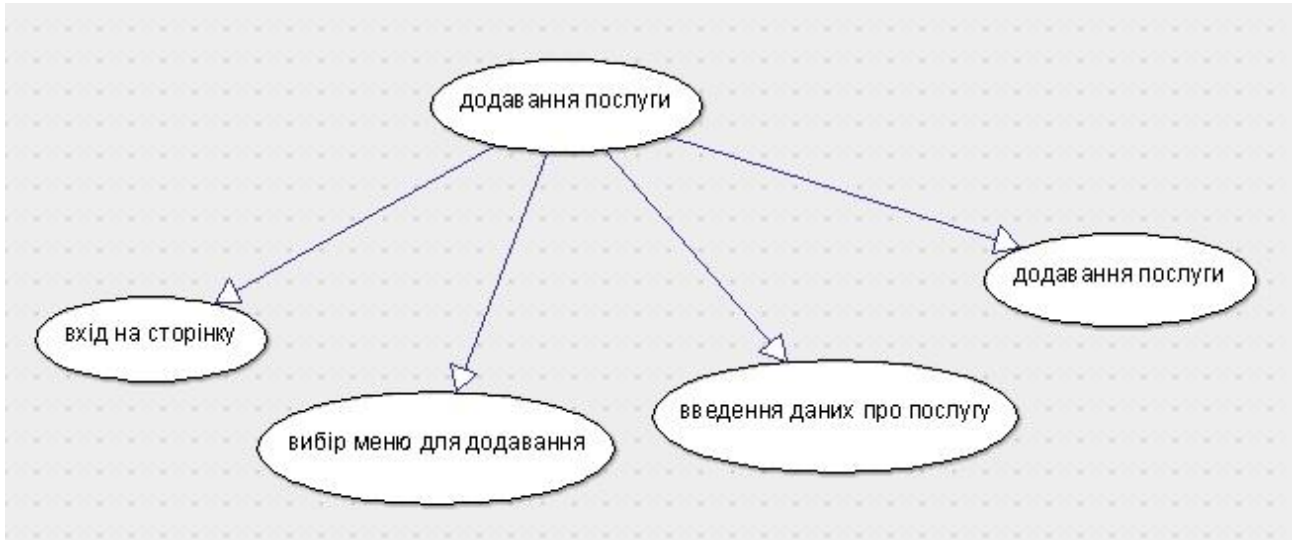


Рис. 1.3 – Діаграма функції процесу додавання послуги

Додавання послуги в веб-сервісі складається з наступних кроків:

1. Вхід на сторінку адміністратора.
2. Введення даних про послугу.
3. Підтвердження додавання.

Характеристику бізнес-процесу додавання послуги наведено в таблиці 1.1

Таблиця 1.1

Характеристика бізнес-процесу додавання послуги

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Додавання послуги на веб-ресурсі
Основні учасники	Адміністратор
Вхідна подія	Запит на додавання послуги
Вхідні документи	Розпорядження директора про внесення нової послуги в перелік.
Вихідна подія	Дані про додавання

Вихідні документи	Повідомлення
Клієнт бізнес-процесу	Адміністратор

На рисунку 1.4 бачимо діаграму функцій процесу видалення послуги.

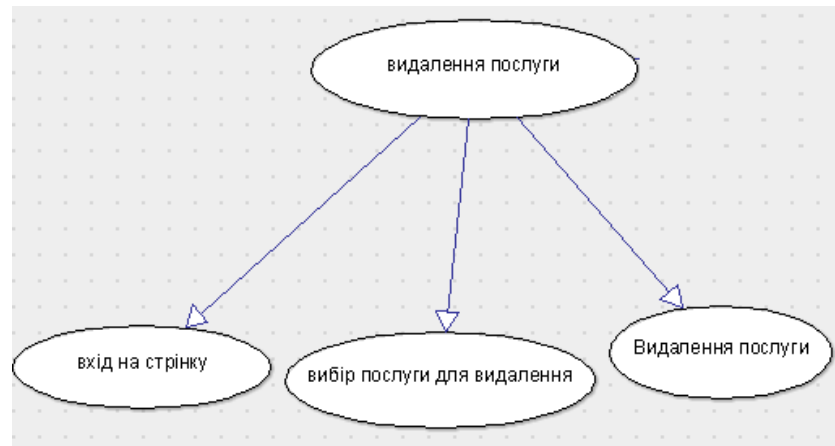


Рис. 1.4 – Діаграма функцій процесу видалення послуги

Видалення послуги в веб-сервісі складається з наступних кроків:

1. Вхід на сторінку адміністратора.
2. Вибір послуги для видалення.
3. Підтвердження видалення.

Характеристику бізнес-процесу видалення послуги наведено в таблиці 1.2

Таблиця 1.2

#### Характеристика бізнес-процесу видалення послуги

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Видалення послуги
Основні учасники	Адміністратор
Вхідна подія	Запит на видалення
Вхідні документи	Розпорядження директора про видалення послуги з переліку послуг.
Вихідна подія	Видалено послугу
Вихідні документи	Повідомлення, з підтвердженням видалення.

На рисунку 1.5 бачимо діаграму функцій процесу авторизації.

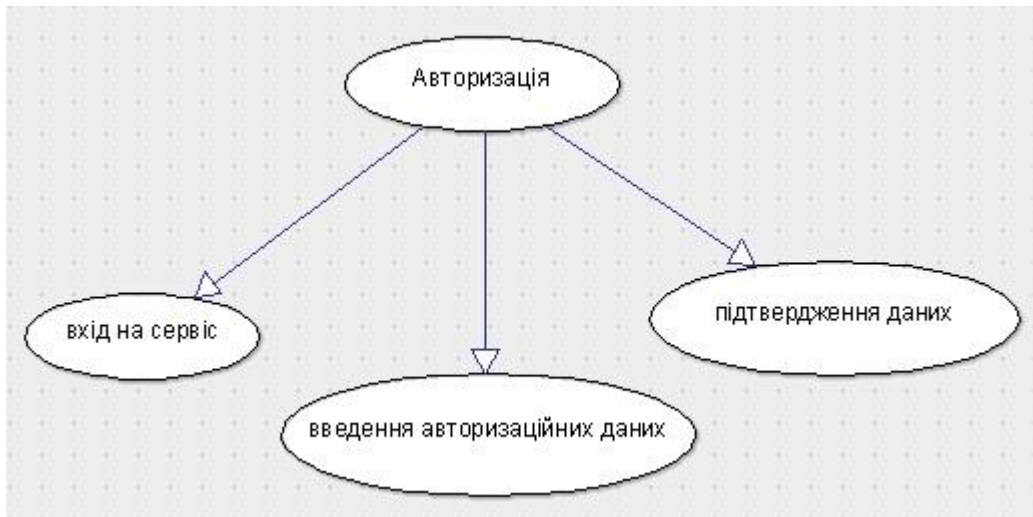


Рис. 1.5 – Діаграма функцій процесу видалення послуги

Авторизація на веб-сервісі складається з наступних кроків:

1. Вхід на сервіс.
2. Введення авторизаційних даних.
3. Підтвердження даних.

Характеристику бізнес-процесу видалення послуги наведено в таблиці 1.3

Таблиця 1.3

#### Характеристика бізнес-процесу авторизація в системі

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Авторизація в системі
Основні учасники	Адміністратор
Вхідна подія	Запит на авторизацію
Вхідні документи	Логін пароль
Вихідна подія	Авторизацію виконано
Вихідні документи	-
Клієнт бізнес-процесу	Адміністратор

На рисунку 1.6 бачимо діаграму функцій процесу редагування послуги.

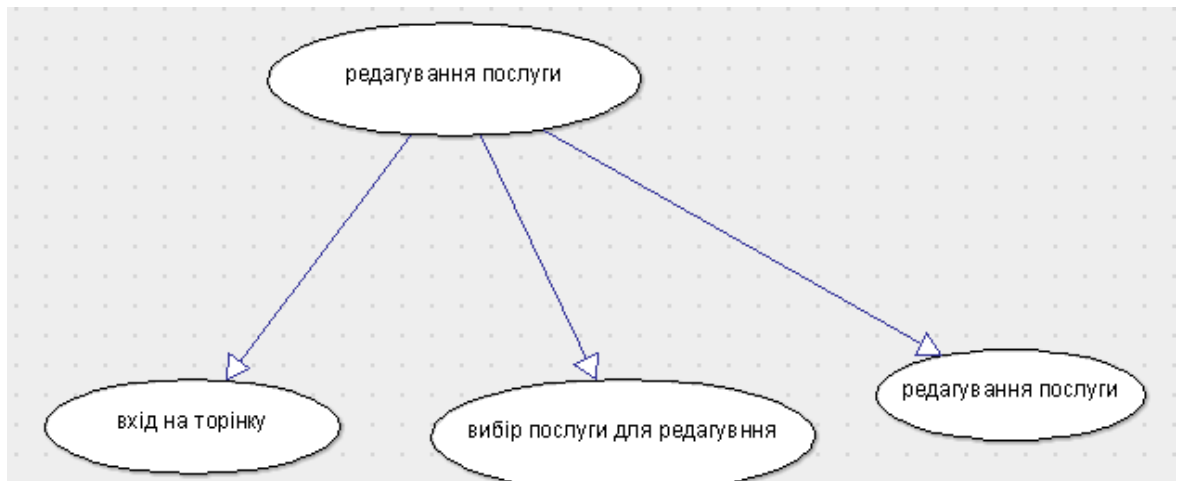


Рис. 1.6 – Діаграма функцій процесу редагування послуги

Редагування послуги складається з наступних кроків:

1. Вхід на сторінку адміністратора.
2. Вибір послуги для редагування .
3. Редагування послуги.

Характеристику бізнес-процесу редагування послуги наведено в таблиці 1.4.

Таблиця 1.4

Характеристика бізнес-процесу редагування послуги

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Редагування послуги
Основні учасники	Адміністратор
Вхідна подія	Запит на редагування послуги
Вхідні документи	Розпорядження директора про редагування послуги
Вихідна подія	Послугу відредаговано
Вихідні документи	Повідомлення про відредагування послуги
Клієнт бізнес-процесу	Адміністратор

На рисунку 1.7 наведено діаграму функцій процесу відправки повідомлення клієнтом адміністратору.

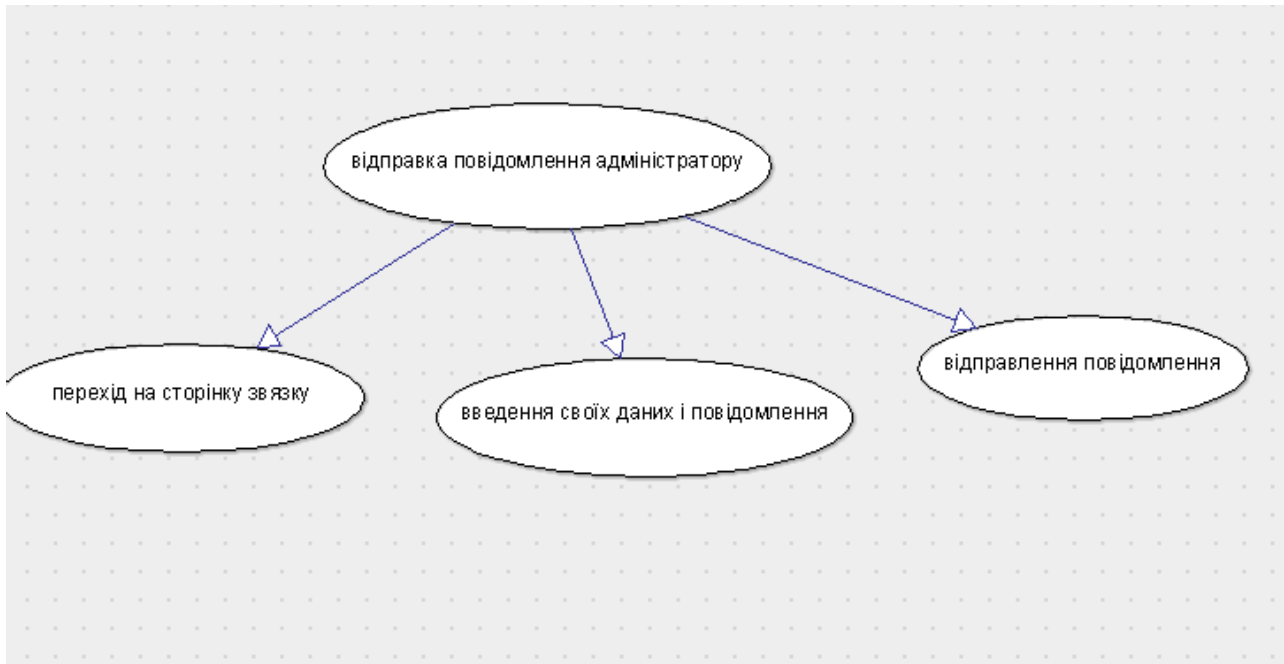


Рис. 1.7 – Відправка повідомлення адміністратору

Відправка повідомлення адміністратору складається з наступних кроків:

- 1) Користувач переходить на сторінку.
- 2) Вносить свої дані і повідомлення у відповідні поля.
- 3) Натискає кнопку відправлення повідомлення.

Характеристику бізнес-процесу «Відправка повідомлення адміністратору» наведено в таблиці 1.5

Таблиця 1.5

Характеристика процесу «Відправка повідомлення адміністратору»

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Відправка повідомлення
Основні учасники	Клієнт
Вхідна подія	Запит на відправку повідомлення
Вхідні документи	Текст повідомлення
Вихідна подія	Надіслане повідомлення адміністратору

Вихідні документи	Надіслане повідомлення повідомлення
Клієнт бізнес-процесу	Клієнт

На рисунку 1.8 наведено діаграму функцій процесу запису на послугу

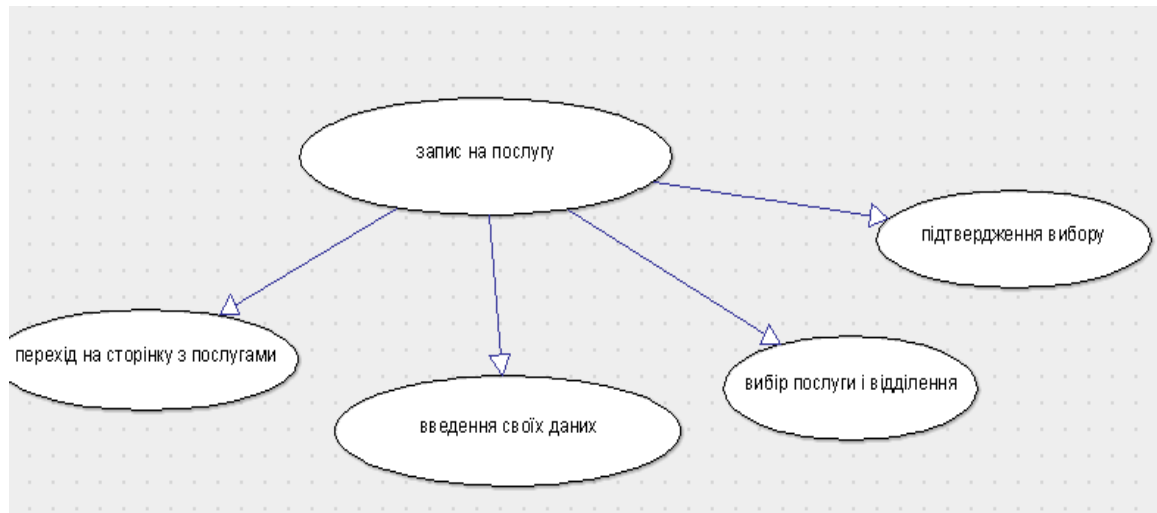


Рис. 1.8 – Процесу запису на послугу

Процес запису на послугу складається з наступних кроків:

- 1) Клієнт переходить на сторінку.
- 2) Вносить свої дані.
- 3) Вибирає послугу і відділення.
- 4) Натискає кнопку відправки замовлення.

Характеристику бізнес-процесу запису на послугу наведено в таблиці 1.5

Таблиця 1.5

#### Характеристика запису на послугу

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Запис на послугу
Основні учасники	Клієнт
Вхідна подія	Запит на відправку повідомлення
Вхідні документи	дані про клієнта та послугу

Вихідна подія	Лист клієнту
Вихідні документи	дані повідомлення
Клієнт бізнес-процесу	клієнт

### 1.3. Огляд та аналіз існуючих аналогів

На етапі проектування розглянемо існуючі програмні продукти, які виконують надання послуг спортивно-оздоровчого центру, а саме:

- Prairie stone sport & wellness center;
- Peterborough Sport & Wellness Centre;
- Sports & Wellness.

Prairie stone sport & wellness center – програмний продукт який ознайомлює нас з спортивним центром «Prairie stone». Даний веб сервіс надає доступну інформацію про центр, послуги, які він надає. Користуватися ним зручно і легко, все розміщено по відповідних меню тому зорієнтуватися легко. Присутній зручний пошук по сайту. Також ресурс дозволяє відправляти повідомлення до адміністратора(рис. 1.10).

Головна сторінка сайту містить наступні елементи:

#### 1. Навігаційне меню сайту (рис. 1.9):

- Welcome;
- programs;
- amenities;
- group fitness;
- membership;
- kids korner;
- join our elist;
- contact us.



Рис. 1.9 – Навігаційне меню сайту

Сторінка для зв'язку з адміністратором

### Contact Us +

Thank you for visiting prairiestonesports.com. If you have any questions or comments please contact by filling out the form below. For immediate attention call 847.285.5405 for more information.

First Name \*

Last Name \*

E-mail Address \*

Phone

Street Address

City

State

Zip Code

Comments/Questions \*

Рис. 1.10 – сторінка Contact Us

2. Слайдер з фотографіями даного центру, календар і короткі відомості (рис. 1.11).

**TENNIS**  
Enjoy a casual game or quench your competitive passion on our indoor tennis courts. Lessons and programs for adults and children available throughout all seasons.

**SCHEDULE** >  
Sign up today

**AQUATIC**  
Train for your next triathlon in our 25 yard lap pool or enjoy a recreational swim/golf with your family in our Activity Pool all within in our spacious Aquatic Center.

**JOIN US** >  
Sign up today

**Meet Your Goals. Your Way.** +

As a member of Prairie Stone Sports & Wellness Center you will find more than just a place to exercise. You will find a family of dedicated employees committed to helping you achieve your wellness goals. It is because of that dedication that you will find state-of-the-art fitness equipment, innovative group exercise classes, superior cleanliness, and the desire to continuously improve our existing ways help you improve your health.

**FITNESS+CALENDAR**

May 2016

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Рис. 1.11 – Головна сторінка сайту з інформацією



3. Поле для пошуку розміщено, на рисунку 1.12:



Рисунок 1.12 – Поле для пошуку

На рисунку 1.13 зображено загальний інтерфейс програмного продукту.

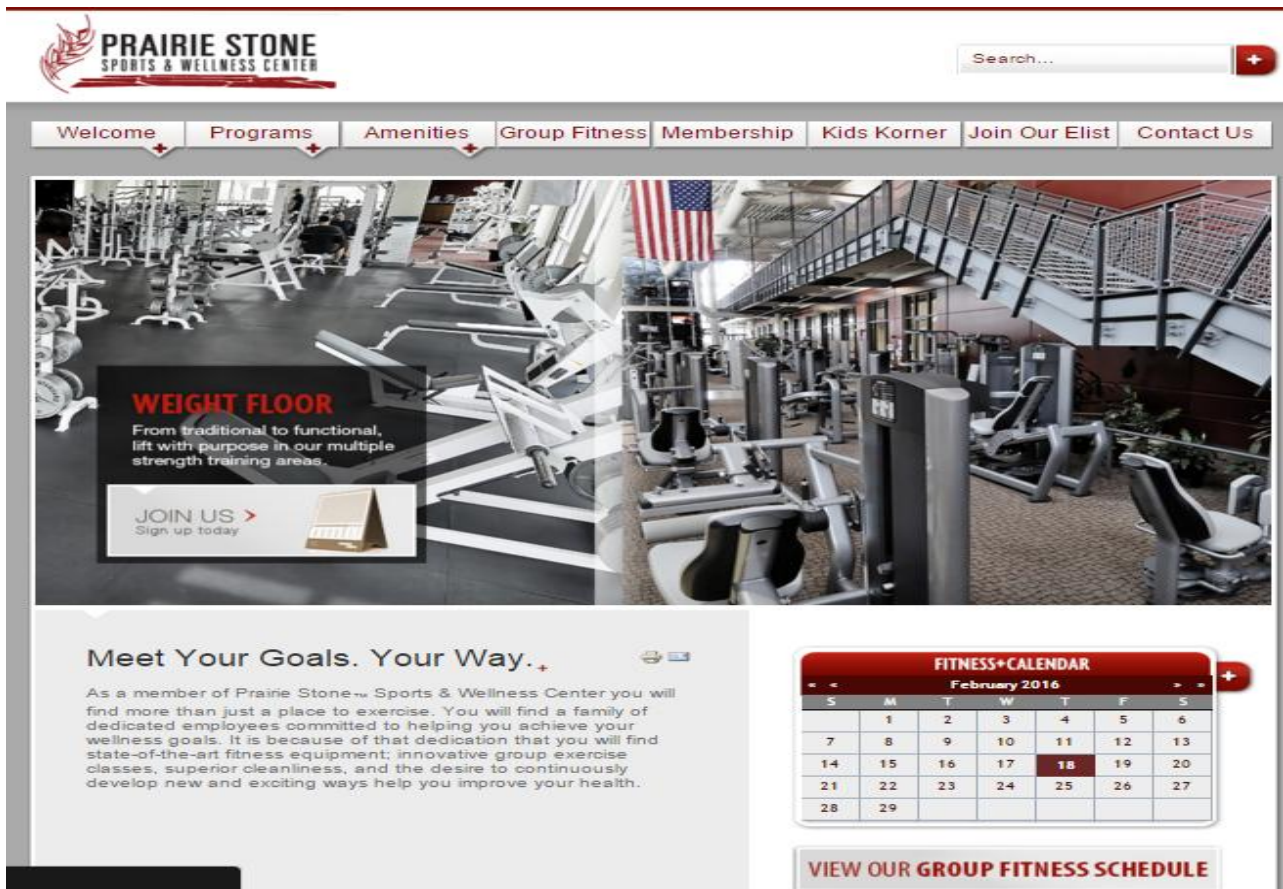


Рис. 1.13 – загальний інтерфейс програмного продукту «Prairie stone»

Peterborough Sport & Wellness Center– програмний продукт який ознайомлює з спортивним центром «Peterborough Sport». Даний web-сайт надає доступ до інформації, яка дає повідомляє користувачу про центр та його послуги. Web-сайт є зручним у користуванні, інформативним і інтуїтивно зрозумілим.

Головна сторінка сайту містить наступні елементи:

1) Навігаційне меню сайту (рисунок 1.14).

- Home;
- membership;
- aquatics;
- fitness;
- personal training;
- child;
- about us.

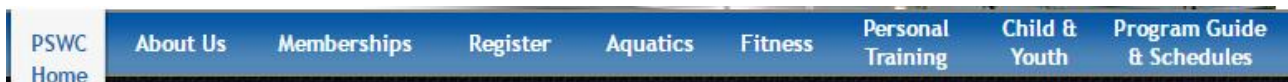


Рис. 1.14 – меню сайту

2) На головній сторінці розміщено інформацію про робочий час центру та інша необхідна користувачу інформація.



Рис. 1.15 – Інформація для користувача



## 1) Навігаційне меню:

- Fitness & wellness
- Amenities
- Family
- Membership
- Club community

## 2) Розклад занять

## 3) Фото з тренувань

Інформація подається коротко, зрозуміло і по темі. Даний сервіс має свою спільноту, що дозволяє краще співпрацювати з клієнтами.

На рисунках 1.18 зображено головну сторінку web-сайту.

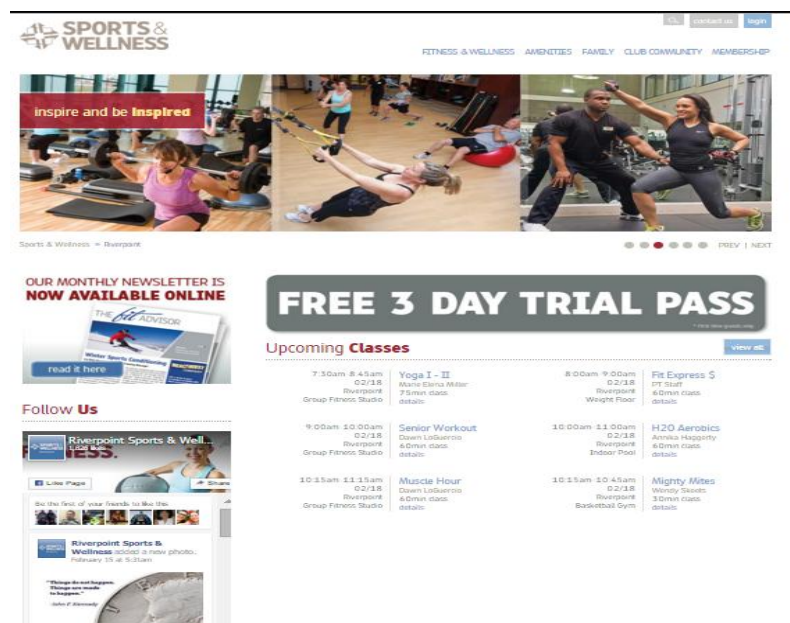


Рис 1.18 – інтерфейс веб-сервісу «Sports & Wellness»

На рисунку 1.19 зображено сторінку з навігацією по web-сайту



[contact us](#)
[login](#)

[FITNESS & WELLNESS](#)
[AMENITIES](#)
[FAMILY](#)
[CLUB COMMUNITY](#)
[MEMBERSHIP](#)

Рис. 1.19 – навігація по веб-сервісу

На рисунку 1.20 зображено сторінку авторизації на сайті.

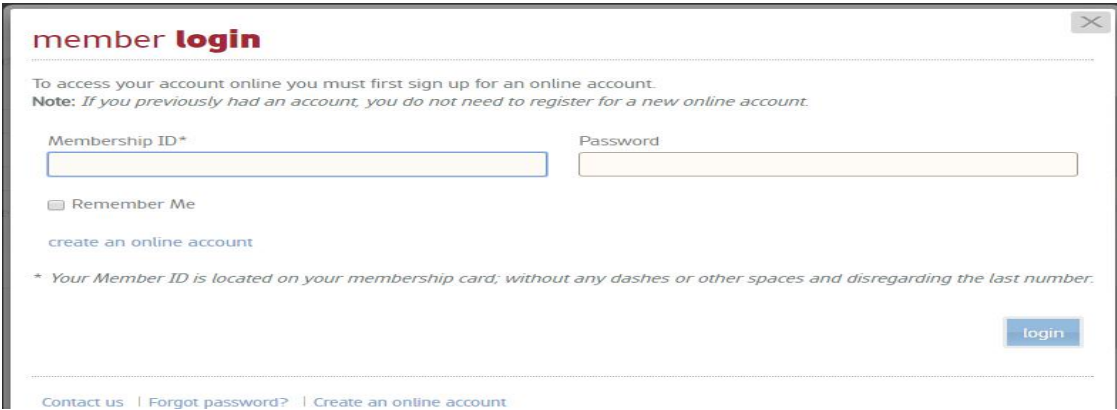


Рис. 1.20 – Сторінка авторизації на сайті

На рисунку 1.21 зображено сторінку зв'язку з адміністратором.

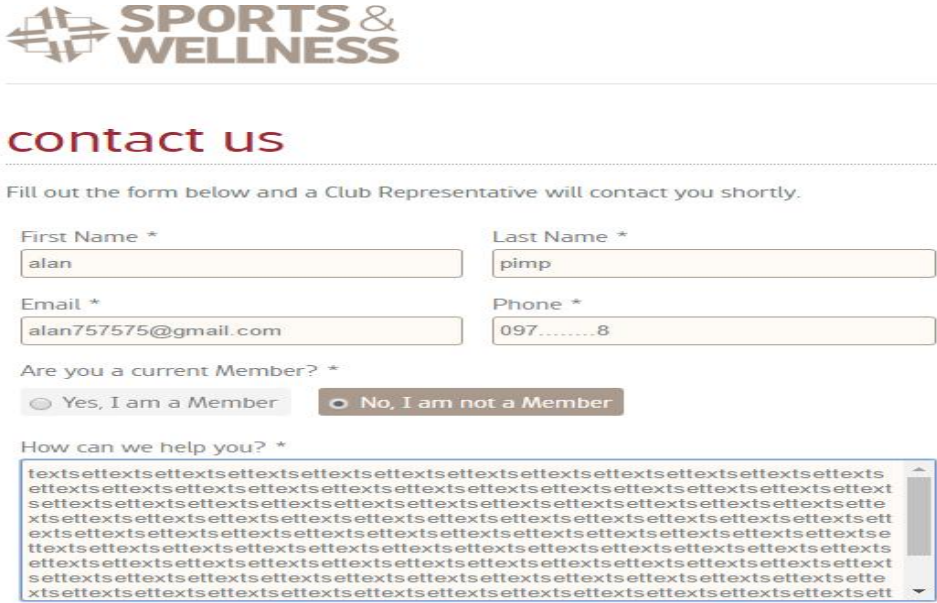


Рис. 1.21 – Сторінка “contact us”

Порівняльна характеристика програмних продуктів відображена в таблиці 1.5

Таблиця 1.5

## Порівняльна характеристика програмних продуктів

Фірма розробник	-	-	-
Назва програмного продукту	Prairie stone	Peterborough Sport & Wellness Centre	Sports & Wellness
Версія продукту	1	1	1
функціональність	Перегляд інформації, зв'язок з адміністратором	Перегляд інформації, зв'язок з адміністратором, пошук по сайту	Перегляд інформації, зв'язок з адміністратором, пошук по сайту, реєстрація авторизація на сайті.і
Інтерфейс користувача	Зручний у використанні	Зручний у використанні інтуїтивно зрозумілий інтерфейс	Зручний у використанні
Допомога користувачу	Присутня	Присутня	Присутня

В результаті аналізу існуючих аналогів, виділено наступні недоліки:

- 1) не адаптовані під об'єкт управління;
- 2) велика кількість інформації;
- 3) скупчення функціоналу;
- 4) відсутність можливості записатися на послугу через веб-сайт.

Кожен з розглянутих продуктів має також свої переваги і недоліки. Проаналізувавши вище переглянуті приклади, можна використати деякі їхні рішення при розробці програмного продукту. Проте потрібно їх адаптувати під обрану предметну область і функціонал розроблюваного продукту.

#### 1.4. Специфікація вимог до програмного продукту

У системі є два актори: адміністратор і клієнт. Згідно їхніх ролей система буде надавати їм відповідний функціонал. Функціональні можливості продукту будуть відображені на діаграмі варіантів використання зображеній на рисунку 1.23.

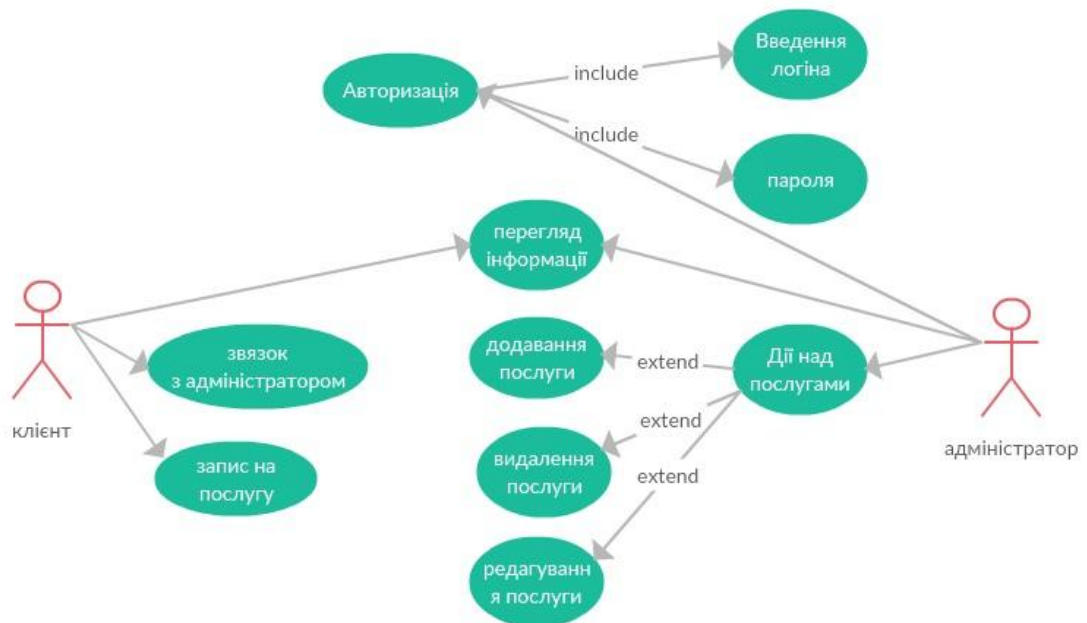


Рис. 1.22 Діаграма варіантів використання

Опис варіантів використання подано у таблицях 1.8-1.11.

Розглянемо такі варіанти використання:

Таблиця 1.8

#### Варіант використання «Перегляд інформації»

Контекст використання	дозволяє переглядати інформацію яку надає ресурс
Дійові особи	Адміністратор, користувач

Передумова	
Тригер	Перший запуск додатку
Сценарій	Вхід користувача на сторінку
Постумова	

На рисунках 1.23 – 1.26 можна переглянути розкадровку варіантів використання.

## Наші послуги

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



## Послуги

▼ Назва	▼ ціна
Масаж	150
тренажерний зал	120-150
групові заняття	120

Рис. 1.23 – Розкадровка варіанту використання «Перегляду інформації»



## Варіант використання «Додавання нової послуги»

Контекст використання	Додавання нової послуги в систему
Дійові особи	Адмін
Передумова	
Тригер	Адмін авторизований
Сценарій	1 – Авторизація на сайті 3 – вибір відповідного меню 2 – Натискання кнопки підтвердження додавання
Постумова	Додається послуга

**Послуги**

Перегляд послуг

Назва послуги	<input style="width: 90%;" type="text"/>
Опис	<input style="width: 90%;" type="text"/>
Ціна	<input style="width: 90%;" type="text"/>
<input style="width: 50%; padding: 5px 20px;" type="button" value="зберегти"/>	

Рис. 1.24 – розкадровка варіанту використання «Додавання послуги»

Варіант використання «видалення послуги»

Контекст використання	видалення послуги
Дійові особи	Адмін
Передумова	
Тригер	Адмін авторизований
Сценарій	1 – Авторизація на сайті 3 – вибір відповідного меню 2 – Натискання кнопки підтвердження видалення
Постумова	Видаляється послуга

номер	назва	опис	ціна	редагувати	видалити	видалити
1	масаж	ТЕКСТТЕКСТТЕКСТТЕКСТ	120	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="button" value="видалити"/>

Рис. 1.25– розкадровка варіанту використання «видалення послуги»

Таблиця 1.11

Варіант використання «зв'язок з адміністратором»

Контекст використання	Відправка повідомлення адміністратору
Дійові особи	клієнт
Передумова	
Тригер	Введено всі необхідні дані
Сценарій	1 – заповнено всі необхідні поля

	2 – Натискання кнопки відправлення повідомлення
Постумова	Відправляється повідомлення

Імя

Email

Телефон

Лист

Надіслати

Рис. 1.26 – розкадровка варіанту використання «зв'язок з адміністратором»

Специфікація функціональних вимог наведена у таблиці 1.12

Таблиця 1.12

## Специфікація функціональних вимог

Ідентифікатор вимог	Назва вимоги	Атрибути вимог		
		Пріоритет	Складність	Контакт
1	Авторизація користувача	Обов'язкове	Середня	-
3	Додавання послуги	Обов'язкове	Середня	-
4	Редагування послуги	Обов'язкове	Середня	-
5	Видалення послуги	Обов'язкове	Середня	-
6	Запис на послугу	Обов'язкове	Висока	-
7	Зворотній зв'язок	Обов'язкове	Висока	-

Специфікація нефункціональних вимог наведена у таблиці 1.9

Таблиця 1.9

## Специфікація нефункціональних вимог

	Назва вимоги	Характеристики
1	Застосовність	
1.1	Час для навчання користувачів	Не більше 45 хв
1.2	Вимірюваний час відгуку для типових завдань	Не більше 30 с
2	Надійність	

Продовження таблиці 1.9

2.1	Середній час безвідмовної роботи	100 год
2.2	Середнє напрацювання до ремонту	1.5 року
2.3	Максимальна норма помилок або дефектів	50
3	Робочі характеристики	
3.1	Місткість (максимальне значення)	2000 користувачів
4	Проектні обмеження	
4.1	Мова програмування	PHP, JavaScript
5	Вимоги до документації, призначеної для користувача, і до системи допомоги	Наявність інструкції користувача
6	Інтерфейси	
6.1	Інтерфейс користувача	Веб-додаток
6.2	Програмні інтерфейси	Бібліотека jQuery RDBMS MySQL PhpMyAdmin
6.3	Комунікаційні інтерфейси	Доступ до мережі інтернет

#### Висновок до першого розділу

У даному розділі було проаналізовано предметну область розроблюваного продукту. Розглянута структура об'єкту дослідження, досліджені актори та їх можливості в системі. Були проаналізовані переваги та недоліки програмних продуктів з аналогічною предметною областю. Інформація отримані при аналізі будуть враховані при розробці програмного продукту.

## РОЗДІЛ 2. ПРОЕКТУВАННЯ

### 2.1 Розробка архітектури програмної системи

Для розробки архітектури інтерактивного web-сервісу «Мережа спортивно оздоровчих центрів «Fitness»» було вирішено використовувати клієнт-серверну архітектуру. Обрана архітектура клієнт – сервер широко застосовується при роботі з базами даних у мережі і передбачає взаємодію та обмін даними між ними. Архітектура клієнт-сервер передбачає такі три основні компоненти:

- набір серверів, які надають інформацію або інші послуги програмам, які звертаються до них;
- набір клієнтів, які використовують сервіси, що надаються серверами;
- мережа, яка забезпечує взаємодію між клієнтами та серверами.

Обробка та збереження даних відбувається на стороні сервера, відображення даних і надсилання запитів на сервер виконується на стороні клієнта. На рисунку 2.1 зображено трирівневу схему архітектури web-додатку.

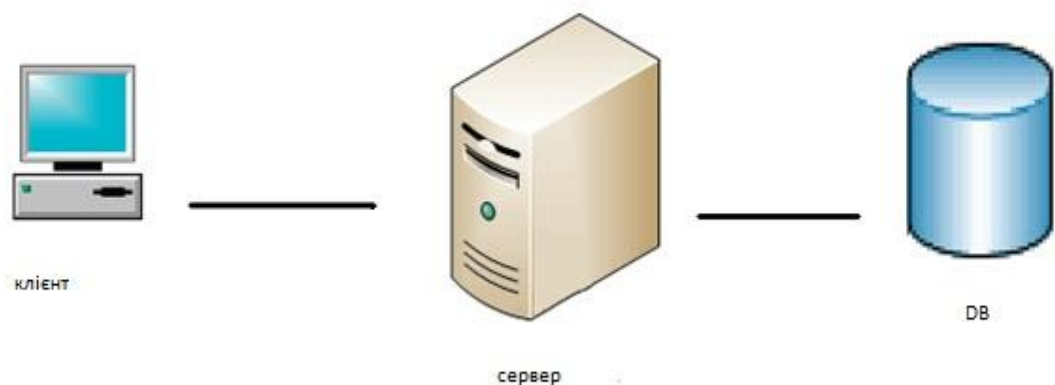


Рис. 2.1 – Клієнт серверна архітектура web-додатку

Перший рівень – це клієнт, який відсилає запити на сервер та приймає результат обробки запитів. Клієнтом є браузер користувача.

Другий рівень – бізнес-логіка додатку. Це логіка за якою web-сервер обробляє отримані від клієнта запити.

Третій рівень – це сама СУБД, яка отримує запити від сервера і повертає потрібні дані на сервер або зберігає їх. Діаграма класів відображає класи спроектованої системи і відношення між ними. Для відображення основної бізнес-логіки розроблюваної системи побудовано UML діаграму класів, яка зображена на рисунку 2.2.

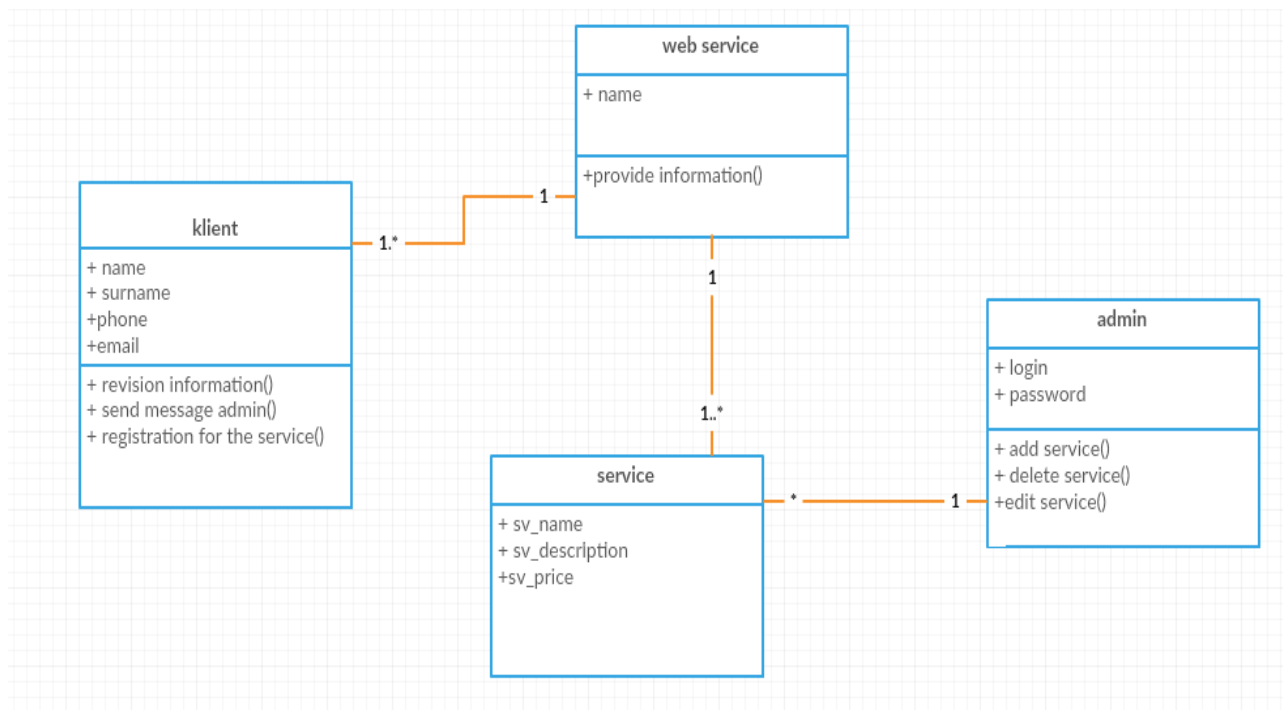


Рис. 2.2 – діаграма класів

Для того щоб знати в якому стані перебуває система потрібно використовувати діаграму станів.

Діаграма станів представляє собою те, що кожен об'єкт системи, який володіє певною поведінкою, може знаходитися в певних станах, переходити із стану в стан, здійснюючи певні дії в процесі реалізації сценарію поведінки об'єкту. На діаграмах 2.3-2.5 зображено діаграми станів для основних сценаріїв.

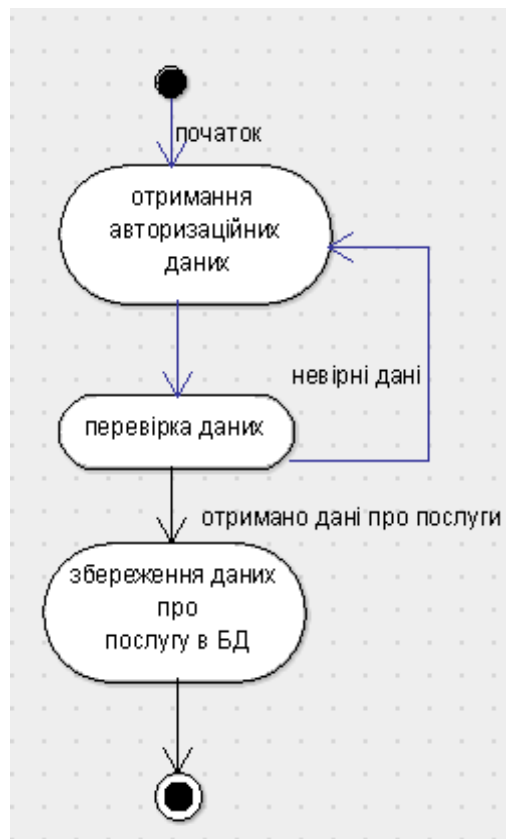


Рис. 2.3 - Діаграма станів для процесу дії над послугами

Система отримує авторизаційні дані від адміністратор та перевіряє їх. Після успішної авторизації система очікує на дані про послуги, перевіряє їх та зберігає в базі даних.





Рис. 2.4 – Діаграма станів для процесу запису на послугу

Система отримує введені дані клієнта. Клієнт обирає послугу та відділення і підтверджує надсилання листа.

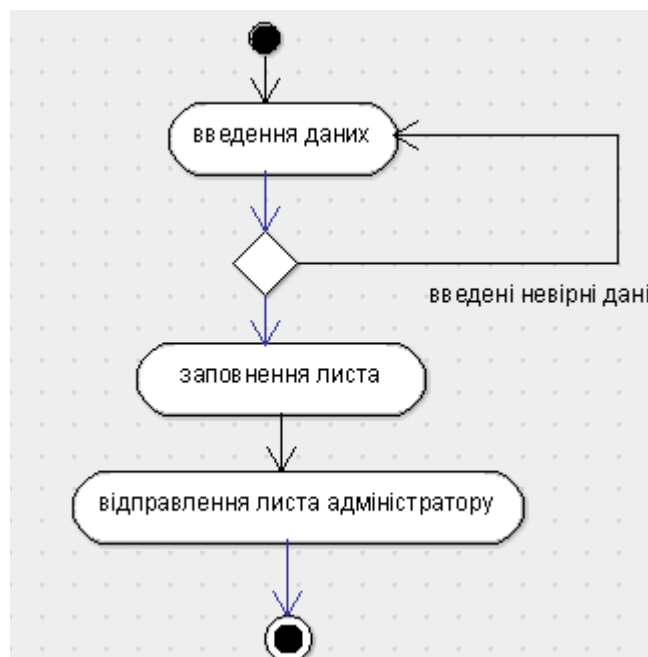


Рис. 2.5 – Діаграма станів для процесу зв'язку з адміністратором

Клієнт вводить свої дані, система перевіряє чи коректно вони введені. Після введення своїх даних клієнт вводить текст листа. Ввівши текст листа клієнт надсилає його адміністратору.

## 2.2 Проектування структури бази даних

Для розробки моделі бази даних обрана реляційна модель даних. Вона найкраще підходить для вирішення цієї задачі, адже вона має ряд наступних переваг:

незалежність програм від даних. Ідея використання баз даних та систем управління базами даних передбачає використання додаткового рівня між прикладними програмами та власне даними, завдяки чому прикладні програмісти можуть абстрагуватися від реалізації самої бази даних, а зосередити свою увагу на логіці обробки даних;

простота розробки та моделювання інформаційного ресурсу як плата за деякі обмеження та уніфікацію на рівні реалізації операцій над даними;

наявність умов керування даними за допомогою операцій над множинами.

В процесі проектування структури бази даних потрібно створити діаграму корпоративної моделі даних.

В діаграмі корпоративної моделі даних графічно представлені зв'язки між об'єктами, з яких складається система. Діаграма корпоративної моделі даних представлена на рисунку 2.6.



Рис. 2.6 діаграма корпоративної моделі даних

ERD (Діаграма сутність-зв'язок) включає сутність і взаємозв'язки, що відображають основні бізнес-правила предметної області. Така діаграма не дуже деталізована, в неї включаються основні сутності і зв'язки між ними, які задовольняють основним вимогам, що пред'являються до системи. ER – діаграма розроблюваного сервісу зображена на рисунку 2.7.

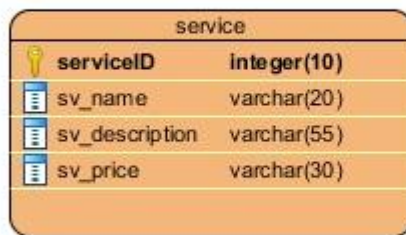


Рис. 2.7 – ERD

На ER – діаграмі зображено наступне відношення: service.

Функціональна модель предметної області бази даних представлена у вигляді діаграми потоків даних.

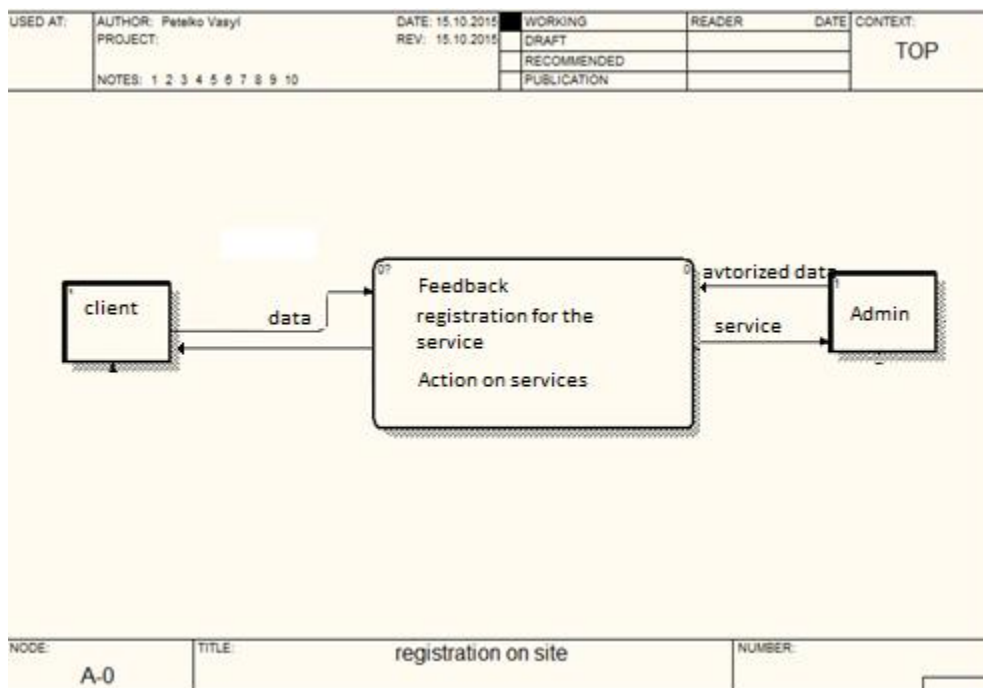


Рис..2.8 - Модель DFD.

На рисунку 2.9 зображено діаграму декомпозиції процесу роботи з послугою.

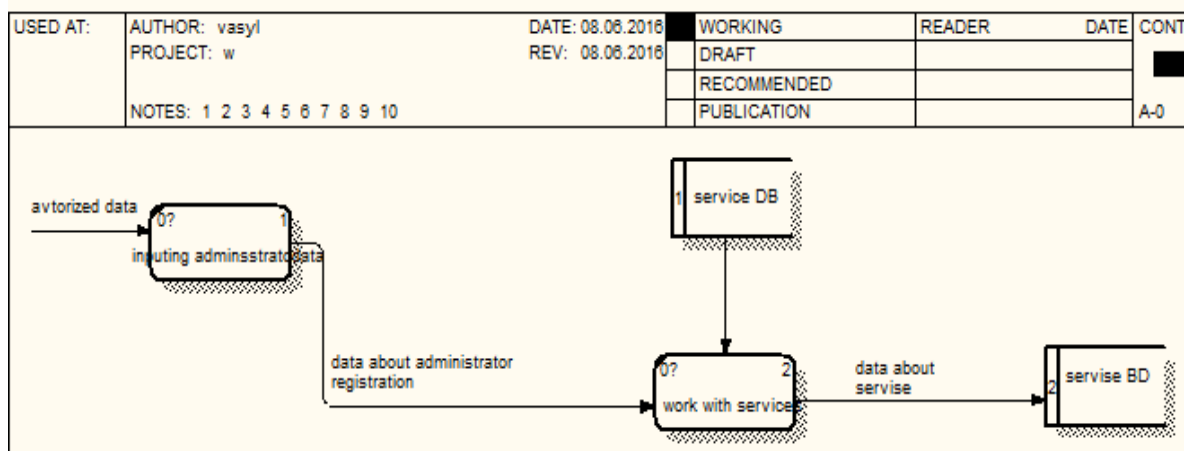


Рис..2.9 - Діаграма декомпозиції процесу роботи адміністратора з послугами.

На рисунок 2.9 видно що на вхід до моделі подається один потік даних і один на виході. На вхід подаються авторизаційні дані користувача системи. Після проходження авторизації починається процес дій над послугами.

У фізичне проектування бази даних входить створення таблиці у відповідній БД, згідно ER - діаграми. Для цього було обрано СУБД MySQL. Нижче наведено DDL код для створення таблиць в БД.

```

CREATE TABLE service
(serviceID integer(10) NOT NULL AUTO_INCREMENT,
sv_name varchar(20) NOT NULL,
sv_description varchar(55),
sv_price varchar(30),
PRIMARY KEY (serviceID)
);
  
```

Висновок до другого розділу.

У даному розділі була розроблена архітектура web-додатку. Спершу було обрано одну із загальних архітектур розробки програмного забезпечення, яка найкраще підходила для вирішення поставленої задачі. Для кращого розуміння архітектури продукту було розроблено діаграми, які показують взаємодію користувачів та їх функцій між собою та у середині системи.

Проаналізовано потоки даних у системі. Усі ці потоки були представленні на діаграмі потоків даних.

## РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ

### 3.1 Програмна реалізація проекту

Розроблювана система є веб-орієнтованою, для її реалізації використовувалися мови програмування: PHP - скриптова мова, яка використовується для розробки веб додатків.

JavaScript - динамічна, об'єктно-орієнтована мова програмування, яка застосовується для розробки web додатків, мобільних додатків і т.д.

JavaScript бібліотека – jQuery.

При створенні шаблону використовувалася мова гіпертекстової розмітки HTML5.

Для надання шаблону завершеного вигляду була використана CSS3 – мова опису зовнішнього вигляду документа. Першим етапом розробки web-сервісу було створення його шаблону.

Було зроблено головну сторінку сервісу, сторінку портфолію, сторінку з інформацією про послуги.

Приклад розмітки головної сторінки наведено нижче.

Підключення стилів файлу стилів:

```
<link href="layout/styles/layout.css" rel="stylesheet" type="text/css" media="all">

<li class="one_third">
  <article class="overlay"><a href="#"><i class="fa fa-line-chart"></i></a>
  <h6 class="heading">Центр на вулиці За Рудкою</h6>
  <p>Знаходимось:</p>
  <p>м.Тернопіль вул. За Рудкою</p>
  <p>16/б</p>
  <p>+0(98)2548613 </p>
</article>
</li>
</ul>
<!-- ##### -->
</section>
</div>
</div>
<!-- ##### -->
```

```

<div class="wrapper row3">
  <main class="container clear">
    <!-- main body -->

    <div class="group">
      <div class="one_third first"></div>
      <div class="two_third">
        <h2 class="heading">Про Нас</h2>
        <p>

```

Основні принципи нашої роботи: доброзичливе ставлення до гостей і членів клубу, першокласний сервіс і всебічна підтримка наших клієнтів у досягненні бажаного результату на тренуваннях.

```

</p>
  <p class="btmspace-80">Наш клуб є ідеальним для успішних людей котрі цінують свій час. Для вас ми пропонуємо оптимальний перелік послуг для підтримання міцного здоров'я і фізичної форми.

```

```

</p>

```

```

<p>Ми надаємо:</p>

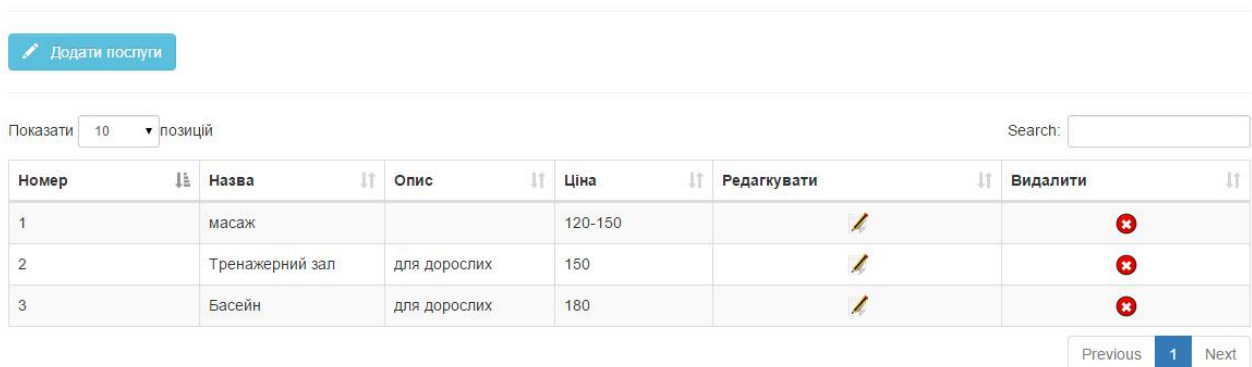
```

Для адміністратора було розроблено адмін панель, результат наведено на рисунку 3.4, за допомогою якої адміністратор, зможе оперувати послугами, які є в базі а саме:

- створювати нову послугу;
- редагувати існуючу послугу;
- видаляти послугу.

На рисунку 3.4 зображено дану панель адміністратора.

## Послуги



Додати послуги

Показати 10 позицій Search:

Номер	Назва	Опис	Ціна	Редагувати	Видалити
1	масаж		120-150		
2	Тренажерний зал	для дорослих	150		
3	Басейн	для дорослих	180		

Previous 1 Next

Рис. 3.1 – Панель адміністратора

За допомогою jquery було додано код в створений файл, який відображає дані з таблиці servise. Код наведено в додатку Е.

Файли які використовувалися для створення функцій адміністратора знаходяться в додатку F.

Для реалізації зворотнього зв'язку було реалізовано алгоритм приведений в додатку G.

Для реалізації запису на послугу було реалізовано алгоритм приведений в додатку H.

### 3.2 Програмна реалізація бази даних.

В базі даних user за допомогою PHPmyAdmin ,було створено таблицю service, для можливості роботи адміністратора з послугами. Ми створюємо такий sql запит:

```
CREATE TABLE IF NOT EXISTS `service` (
  `serviceID` int(11) NOT NULL AUTO_INCREMENT,
  `sv_name` varchar(25) NOT NULL,
  `sv_description` varchar(50) NOT NULL,
  `sv_price` int varchar(5),
  Primary key(`serviceID`)
);
```

Результатом виконання SQL запиту є таблиця зображена на рисунку 3.4.

























#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Дополнительно	Действие
<input type="checkbox"/>	1 <u>serviceID</u>	int(11)			Нет	Нет	AUTO_INCREMENT	     
<input type="checkbox"/>	2 sv_name	varchar(25)			Нет	Нет		     
<input type="checkbox"/>	3 sv_description	varchar(50)			Нет	Нет		     
<input type="checkbox"/>	4 sv_price	varchar(7)			Нет	Нет		     

Рис. 3.4 – таблиця service

Висновки до третього розділу:

В цьому розділі було описано обрані для реалізації web-сайту “Fitness”. Також було реалізовано функціонал системи відповідно до вимог та наведено фрагменти програмного коду.



## Розділ 4. Тестування та дослідна експлуатація

## 4.1 Тестування

Тестування процес який застосовується для перевірки чи відповідає розроблена система заданим специфікаціям. Метою проведення мною функціонального тестування є знаходження невідповідностей між розробленою системою і очікуваним результатом.

Для того, щоб провести функціональне тестування необхідно розробити тестові випадки, що містить такі дані:

- id - ідентифікатор тестового випадку;
  - summary – опис об’єкту тестування;
  - expected result – очікуваний результат;
  - passed/failed – показує чи тест пройшов чи провалився;
  - pre-condition – умова, яка необхідна для виконання, щоб виконати тестовий випадок;
  - steps to reproduce – шлях проходження тестового випадку;
- Було розроблено 6 тестових випадків які успішно були пройдені.

Id	Summary	Steps to reproduce	Pre - condition	Expected result	Passed/Failed
1	Verify ability of user to visit portfolio page.	1. choosing link “Портфоліо і послуги” 2. Choose link and click this linck	1. User visite some page.	1.user click a linck and go to portfolio page	Passed
2	Verify ability of user to visit page with	1. choosing link “Портфоліо і	1. User visite some page..	1.user click a linck and go to	Passed

	service list.	послуги” 2. Choose link and click this linck		portfolio page	
3	Verify ability of user return to home page	1. choosing link “Домашня” 2. Choose link and click this linck	1. User visited some page	1.user click a linck and go to home page	Passed
4	Verify ability of administrator visited admin page	1. Enter valid Login. 2. Enter valid Password 3. Click on Sign in button	1. User is losated on the Log in page. 2. User has an account.	1. User is succesfully logged in to account. 2. System forward admin to the admin page.	Passed
5	Verify ability of administrator add new service	1.Click “add” button 2. enter data about new servise	1.visite the administrator page	1. admin add new service	Passed
6	Verify ability of administrator delete service	1.Click “delete” button 2. choos servise 3.delete	1.visite the administrator page	1. admin delete service	Passed

		choosing servise			
--	--	---------------------	--	--	--

### GUI тестування.

Для того щоб GUI тестування було більш ефективним, використовують автоматизоване тестування інтерфейсу. Автоматизоване тестування проводилося за допомогою програмного продукту Selenium. Розроблений програмний продукт є веб-сервісом, то для тестування було встановлено плагін Selenium IDE для браузеру Mozilla Firefox. Результат проведеного тестування користувацького інтерфейсу за допомогою Selenium IDE представлено на рисунку 4.1.

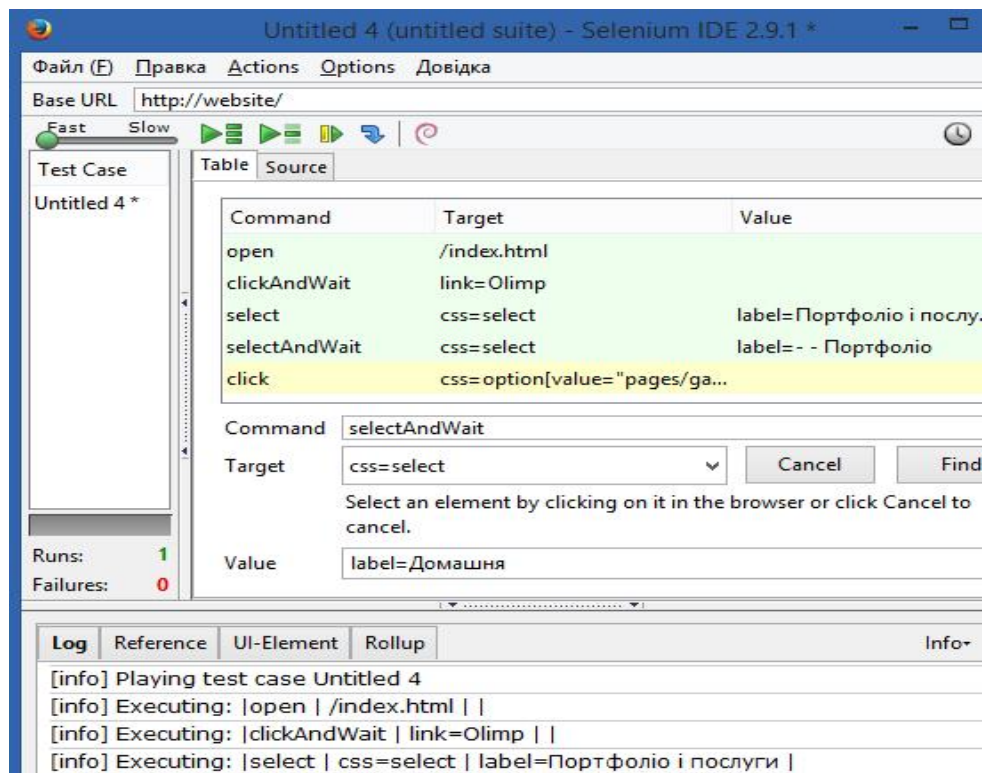


Рис. 4.2 – тестування в Selenium IDE

## 4.2 Розгортання програмного продукту

Для забезпечення повноцінного функціонування інтерактивного веб-сервісу для мережі спортивно оздоровчих центрів, потрібно розмістити його компоненти на веб-сервері, а саме файли сайту та базу даних. Для цього в панелі управління хостингом необхідно створити новий сайт та присвоїти йому доменне ім'я, після чого створити на сервері директорію з назвою яка відповідає доменному імені самі, в ній створити папку www та завантажити всі файли сайт в цю папку.

Для користування сайтом користувачу знадобиться доступ до інтернету з комп'ютера який використовується, та встановлений веб-браузер Opera, Chrome, Safari, Mozilla або інший браузер що підтримує стандарти HTML5 та CSS3.

## 4.2 Інструкція користувача

Для того щоб користуватися даним веб сервісом користувачу потрібен комп'ютер, підключення до мережі інтернет і веб-браузер. Для того щоб переглянути інформацію про центр його робочий графік користувачу потрібно відвідати веб-сайт, і перейти по силці на необхідну сторінку.

**Наші послуги**

**Тренажерний зал**

В клубі представлені різні види професійних тренажерів для кожної групи м'язів. Зона вільних ваг укомплектована широким асортиментом гантелей, грифів і дисків відповідають всім сучасним вимогам. Широкий вибір ваг дозволяє займатися як зовсім так і досвідченому спортсмену.

Заняття в тренажерному залі допоможуть вам покращити всі фізичні якості, схуднути або набрати вагу, зміцнити здоров'я. А наші висококваліфіковані тренери зроблять процес досягнення вашої мети швидким та безпечним!

**Кардіо тренажери**

Кардіо – заняття необхідні для тренування серцево-судинної системи, розвитку витривалості, є незамінним для тих хто прагне покращити свою форму. Кардіозалі нашого залу містить наступні тренажери: – Бігові дорожки (ергономічного дизайну, зручні у користуванні) – Горизонтальний велотренажер (мас лонгу підтримку для спини).

**Масаж**

Це комплексний підхід до здоров'я і краси кожної людини, обіцяє стимулює і зміцнює всі системи організму. Масажи допомагає вам у досягненні цілі. Довірте покладатися про ваше здоров'я і красу професіоналам і просто розслабтесь. Відведітьте своє від кращих спеціалістів міста Львова. Приплив сил, м'язовий тонус і незрівняне відчуття легкості з'являться без найменших зусиль з вашого боку.

**Наші послуги**

120-150грн (Масаж)  
100-150грн (Тренажерний зал)

**Працюємо**

Дні	Час
Понеділок-П'ятниця	9-21
Субота	9-20
Неділя	9-19

Рис. 4.2 – сторінка з послугами і робочим часом

Для того щоб записатися на прийом користувачу необхідно ввести свої дані:

- Ім'я
- Прізвище
- Електрону адресу на яку прийде лист;

Вибрати бажану послугу і центр в якому вона буде надаватися.

Даний процес зображено на рисунку 4.3.

Імя:  
alan

Прізвище:  
pimp

Телефон:  
380983340614

E-mail:  
alan757575@gmail.com

Послуга:  
масаж ▼

Відділення:  
За рудкою 16\6 ▼

Надіслати

Рис. 4.3 – запис на послугу

У випадку якщо користувач хоче зв'язатися з адміністратором він може написати йому лист скориставшись відповідним функціоналом сайту. Для цього потрібно перейти на сторінку «зв'язатися з нами» і заповнити необхідні форми. Результат виконання зображено на рисунку 4.4.

ЗВ'ЯЖІТЬСЯ З НАМИ

ДОМАШНЯ | ПОРТФОЛІО І ПОСЛУГИ | ADMIN | [ЗВ'ЯЗАТИСЯ З НАМИ](#)

Імя:  
alan pimp

E-mail:  
alan757575@gmail.com

Телефон:  
0983340614

Лист:  
hello world

Надіслати

Рис. 4.4 – лист адміністратору

Відвідувач сайту має можливість переглянути фотографії центру. Це надасть йому уявлення про центр, як він виглядає, його послуги, обладнання центру. Портфоліо центру зображено на рисунку 4.5

#### Наше портфоліо

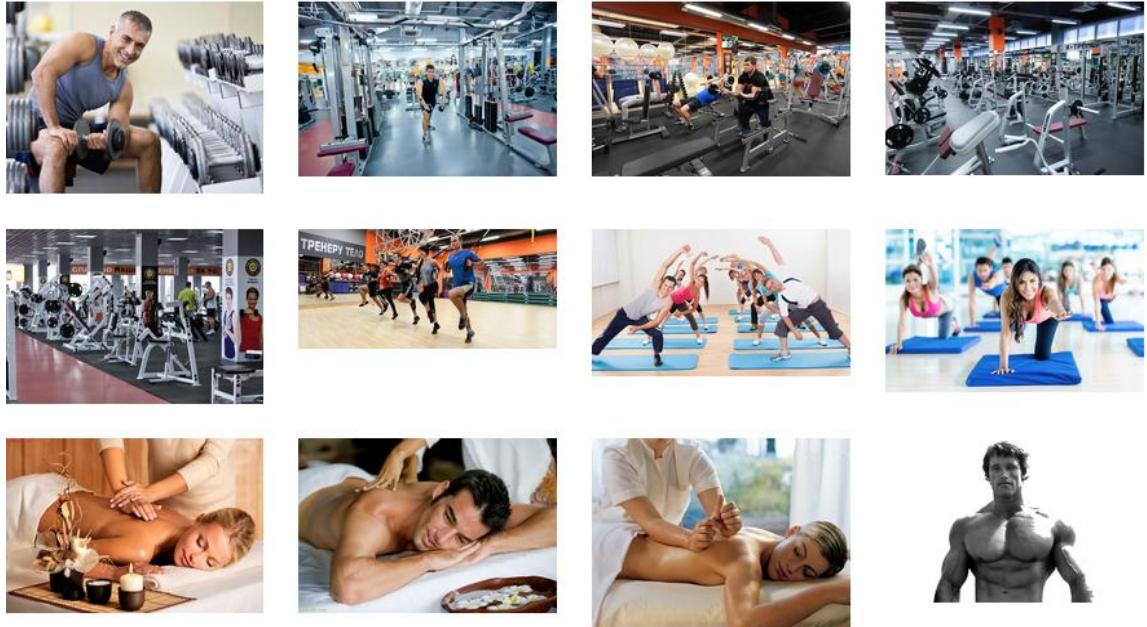


Рис. 4.5 – портфоліо сайту

Щоб оперативно додавати, видаляти послуги адміністратор має свою сторінку з реалізованим інтерфейсом. Що дозволяє йому керувати даними таблиці. На рисунку 4.6 зображена сторінка адміністратора з послугами.

#### Послуги

[Додати послуги](#)

Показати 10 позицій

Номер	Назва	Опис	Ціна	Редагувати	Видалити
1	масаж		120-150		
2	Тренажерний зал	для дорослих	150		
3	Басейн	для дорослих	180		

Previous 1 Next

Рис. 4.6 – сторінка адміністратора

Для того щоб мати можливість адміністратору працювати з послугами він повинен авторизуватися в системі ввівши свій логін і пароль. Процес авторизації зображено на рисунку 4.7.



Вхід

User  
redbool

Password  
.....

Log-In

Рис. 4.7 – авторизація

Пройшовши процес авторизації адміністратор може працювати з послугами а саме: додавати послугу, видаляти послугу, редагувати її.

На рисунку 4.8 зображено процес додавання нової послуги.

Послуги



👁️ Переглянути послуги

Назва послуги	масаж
Опис	-
Ціна	120-150

+ Зберегти цей запис

Рис. 4.8 додавання послуги



## Висновок до четвертого розділу

В даному розділі було проведено функціональне тестування розробленого програмного продукту. Проведено автоматизоване тестування користувацького інтерфейсу з використанням плагіну Selenium IDE для браузера Mozilla Firefox. Що дозволило перевірити розроблений програмний продукт на недоліки. Описано розгортання програмного продукту та створено інструкцію для користувача.

## Список використаної літератури

1. Буч Г. Язык UML: руководство пользователя / Г. Буч, Д. Рамбо, И. Якобсон // [пер. с англ. Н. Мухина]. – Москва. – 2007. – 493 с.
2. Вендров А.М. CASE–технологии. Современные методы и средства проектирования информационных систем. – М.: Финансы и статистика. – 2000.
3. Голицина О. Л. Основы алгоритмизации и программирования /О.Л.Голицина, И.И.Попов// Учебное пособие. - М.: ФОРУМ: ИНФРА-М. – 2005.
4. Дивак М.П. Системний аналіз та проектування КІС /М.П.Дивак// Навчальний посібник – Т.: Економічна думка. – 2004.
5. Мэтт Зандстра. PHP: объекты, шаблоны и методики программирования, 3-е издание = PHP Objects, Patterns and Practice, Third Edition. — М.: «Вильямс», 2010. — С. 560
6. Сеницын С. В., Налютин Н. Ю. Верификация программного обеспечения. — М.: БИНОМ, 2008. — 368 с

## Додаток А

Таблиця А.1

## Глосарій

Термін	Опис терміну
1. Основні поняття та категорії предметної області та проекту.	
Веб-сайт	Сукупність веб-сторінок, доступних у мережі Інтернет, які об'єднані як за змістом, так і за навігацією. Фізично сайт може розміщуватися як на одному, так і на кількох серверах.
База даних	Впорядкований набір логічно взаємопов'язаних <u>даних</u> , що використовуються спільно та призначені для задоволення інформаційних потреб <u>користувачів</u> .
Web-система	Клієнт-серверний додаток, в якому клієнтом виступає браузер, а сервером - веб-сервер.
Реєстрація в системі	Зареєстрований користувач може обрати собі <u>ім'я</u> ( <u>логін</u> ), яким будуть підписуватись в системі всі його редагування.
Послуга	можна виділити два підходи до визначення послуги: 1) "послуга" як дія, що приносить користь, допомогу іншому; 2) "послуга" як продукт (результат) діяльності сервісного

	підприємства.
2. Користувачі системи	
Користувач	Простий користувач який може переглядати вміст сайту(системи), але не має права доступу до більшості функцій системи.
Адміністратор	Має повні права доступу до системи її функціональних можливостей, а також маніпуляції над даними, якими оперує система.
Авторизований користувач	Користувач який має більше прав доступу до функцій системи, але деякий контент для нього закритий.
Upperbody	Тренування для тих у кого проблемні зони знаходяться вище пояса.
TABS	Тренування спрямоване на пропрацювання абдомінальних м'язів живота.
Functional	Вид вправ при якому пропрацьовують всі м'язи тіла.
Streching	Вид аеробіки, який містить комплекс вправ на розтягування.

## Додаток Б

### Специфікація вимог до програмного продукту для web-сервісу “Мережі спортивно оздоровчих центрів”

#### 1. Вступ

##### 1.1 Призначення, мета

Даний програмний продукт розроблений, для надання інформації користувачам про відділення даного центру, послуги які він надає. Надання адміністратору можливості вести облік послуг, проводи з ними маніпуляції.

##### 1.2 Продукти-аналоги

Схожими продуктами до нашої системи є:

- Prairie stone sport & wellness center
- Peterborough Sport & Wellness Centre
- Sports & Wellness center

#### 2. Загальний опис

##### 2.1 Характеристики продукту

Оскільки ПЗ покликане стандартизувати і полегшує адміністрування послуг, воно має в собі певні характеристики та відповідно функції що їх реалізують:

- Додавання послуги.
- Видалення послуги.
- Редагування послуги
- Перегляд інформації

##### 2.2 Класи користувачів та їх характеристики

У системі присутні два класи користувачів адміністратори та користувачі. Функціонал даних користувачів відрізняється різними рівнями доступу та можливостей у системі.

### 2.3 Середовище функціонування

Розроблена система є web-додатком, то вона має коректно функціонувати в усіх популярних веб-браузерах: Safari, Mozilla FireFox, Opera, Internet Explorer, Microsoft Edge, Google Chrome.

## 3. Характеристики системи

### 3.1. Додавання нової послуги

#### 3.1.1 Опис і пріоритет

Функціонал, що дає адміністратору можливість додавати нову послугу яка буде відображатися на сторінці сайту.

#### 3.1.2 Послідовності дія/відгук

- Відкрити панель адміністратора.
- Вибрати «додати послугу»
- Заповнити відповідну форму.
- Натиснути кнопку «Додати».

#### 3.1.3 Функціональні вимоги

REQ-1.1: всі поля обов'язкові для заповнення;

REQ-1.2: відправка інформації про послугу на сторінку

### 3.2. Видалення нової послуги

#### 3.2.1 Опис і пріоритет

Функціонал, що дає адміністратору можливість видалити послугу яка втратила свою актуальність.

#### 3.2.2.Послідовності дія/відгук

- Відкрити панель адміністратора.
- Вибрати «видалити послугу»
- Заповнити відповідну форму.
- Натиснути кнопку «Видалити».

REQ-2.1: всі поля обов'язкові для видалення;

REQ-2.2: адміністратор видаляє послугу.

### 3.3. Редагування послуги

#### 3.3.1 Опис і пріоритет

Функція, яка дає можливість адміністратору редагувати послугу.

#### 3.3.2 Послідовності дій/відгук

- Відкрити адміністративну панель сайту.
- Вибрати «видалити послугу»
- Ввести в форму нову інформацію про послугу.
- Натиснути кнопку «Редагувати»

### 3.4. Запис на послугу

#### 3.4.1 Опис і пріоритет

Функція, яка дає можливість клієнту записатися на послугу.

#### 3.4.2 Послідовності дій/відгук

- Перейти на необхідну сторінку.
- Ввести всі необхідні дані.
- Вибрати зі списку послугу і відділення.
- Натиснути кнопку «Надіслати»

### 3.5. Лист адміністратору

#### 3.5.1 Опис і пріоритет

Функція, яка дає можливість клієнту написати листа адміністратору.

#### 3.5.2 Послідовності дій/відгук

- Перейти на необхідну сторінку.
- Ввести всі необхідні дані.
- Ввести текст повідомлення.
- Натиснути кнопку «Надіслати»



## Додаток В

# Код тестових випадків автоматизованого тестування в Selenium IDE

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head profile="http://selenium-ide.openqa.org/profiles/test-case">
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="selenium.base" href="http://website/" />
<title>New Test</title>
</head>
<body>
<table cellpadding="1" cellspacing="1" border="1">
<thead>
<tr><td rowspan="1" colspan="3">New Test</td></tr>
</thead><tbody>
<tr>
<td>open</td>
<td>/index.html</td>
<td></td>
</tr>
<tr>
<td>clickAndWait</td>
<td>link=Fitness</td>
<td></td>
</tr>
<tr>
<td>select</td>
<td>css=select</td>
<td>label=- - Πορτολίο</td>
</tr>
<tr>
<td>clickAndWait</td>
<td>css=option[value=&quot;pages/gallery.html&quot;]</td>
<td></td>
</tr>
<tr>
<td>selectAndWait</td>
<td>css=select</td>

```

```

    <td>label=- - Послуги</td>
</tr>
<tr>
    <td>click</td>
    <td>css=option[value=&quot;service.php&quot;]</td>
    <td></td>
</tr>
<tr>
    <td>selectAndWait</td>
    <td>css=select</td>
    <td>label=Домашня</td>
</tr>
<tr>
    <td>click</td>
    <td>css=option[value=&quot;../index.html&quot;]</td>
    <td></td>
</tr>
<tr>
    <td>selectAndWait</td>
    <td>css=select</td>
    <td>label=admin</td>
</tr>
<tr>
    <td>click</td>
    <td>css=option[value=&quot;pages/admin/admin.php&quot;]</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
    <td>id=btn-add</td>
    <td></td>
</tr>
<tr>
    <td>type</td>
    <td>name=emp_name</td>
    <td>групові заняття</td>
</tr>
<tr>
    <td>type</td>
    <td>name=emp_dept</td>
    <td>група</td>
</tr>
<tr>

```

```
<td>type</td>
<td>name=emp_salary</td>
<td>182</td>
</tr>
<tr>
<td>click</td>
<td>id=btn-save</td>
<td></td>
</tr>

</tbody></table>
</body>
</html>
```

## Додаток Г

## Лістинг коду авторизації адміністратора

```

<?php
define('DB_HOST', 'localhost');
define('DB_NAME', 'user');
define('DB_USER','root');
define('DB_PASSWORD','');

$con=mysql_connect(DB_HOST,DB_USER,DB_PASSWORD) or die("Failed to connect to MySQL: " .
mysql_error());

$db=mysql_select_db(DB_NAME,$con) or die("Failed to connect to MySQL: " . mysql_error());
/*
$ID = $_POST['user'];
$Password = $_POST['pass'];
*/
function SignIn()
{
session_start(); //starting the session for user profile page
if(!empty($_POST['user'])) //checking the 'user' name which is from Sign-In.html, is it empty or have some
text
{
$query = mysql_query("SELECT * FROM UserName where userName = '$_POST[user]' AND pass =
'$_POST[pass]'") or die(mysql_error());
$row = mysql_fetch_array($query) or die(mysql_error());
if(!empty($row['userName']) AND !empty($row['pass']))
{
$_SESSION['userName'] = $row['pass'];
?>
<a href="pages/admin/admin.php">admin</a>
<? }
else
{
echo "SORRY... YOU ENTERD WRONG ID AND PASSWORD... PLEASE RETRY...";
}}}
if(isset($_POST['submit']))
{
SignIn();
}?>

```

## Додаток Е

## Лістинг коду функцій адміністратора

```

$(".content-loader").load('add_form.php');
    $("#btn-add").hide();
    $("#btn-view").show();
});
});
$("#btn-view").click(function(){

    $("body").fadeOut('slow', function()
    {
        $("body").load('admin.php');
        $("body").fadeIn('slow');
        window.location.href="index.php";
    });
});
</script>
</head>
<body>

    <div class="container">
        <h2 class="form-signin-heading">Послуги</h2><hr />
        <button class="btn btn-info" type="button" id="btn-add"> <span class="glyphicon glyphicon-
pencil"></span> &nbsp; Додати послуги</button>
        <button class="btn btn-info" type="button" id="btn-view"> <span class="glyphicon glyphicon-eye-
open"></span> &nbsp; Переглянути послуги</button>
        <hr />
        <div class="content-loader">
        <table cellpadding="0" cellspacing="0" width="100%" id="example" class="table table-striped table-hover table-
responsive">
            <thead>
                <tr>
                    <th>Номер</th>
                    <th>Назва</th>
                    <th>Опис</th>
                    <th>Ціна</th>
                    <th>Редагувати</th>
                    <th>Видалити</th>
                </tr>
            </thead>

```

```

<tbody>
<?php
require_once 'dbconfig.php';
$stmt = $db_con->prepare("SELECT * FROM service ORDER BY serviseID ");
$stmt->execute();
while($row=$stmt->fetch(PDO::FETCH_ASSOC))
{?>
    <tr>
    <td><?php echo $row['serviseID']; ?></td>
    <td><?php echo $row['sv_name']; ?></td>
    <td><?php echo $row['sv_description']; ?></td>
    <td><?php echo $row['sv_price']; ?></td>
    <td align="center">
    <a id="<?php echo $row['emp_id']; ?>" class="edit-link" href="#" title="Редагувати">
    
    </a></td>
    <td align="center"><a id="<?php echo $row['emp_id']; ?>" class="delete-link" href="#"
title="Видалити"> ...

```

Створюємо файл з налаштуваннями бази даних з PDO розширенням.

```

<?php
$db_host = "localhost";
$db_name = "user";
$db_user = "root";
$db_pass = "";
try{
$db_con = new PDO("mysql:host={$db_host};dbname={$db_name}",$db_user,$db_pass);
    $db_con->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
}
catch(PDOException $e){
    echo $e->getMessage();
}?>

```

Внесення нового запису в таблицю service.

```

<?php
require_once 'dbconfig.php';
if($_POST)
{
    $emp_name = $_POST['sv_name'];
    $emp_dept = $_POST['sv_description'];
    $emp_salary = $_POST['sv_price'];
    try{
        $stmt = $db_con->prepare("INSERT INTO service(sv_name,sv_description,sv_price) VALUES(:ename,
:edescription, :eprice)");
        $stmt->bindParam(":ename", $sv_name);

```

```
$stmt->bindParam(":description", $sv_description);
$stmt->bindParam(":price", $sv_price);
if($stmt->execute())
    {
echo "Successfully Added";
    }
    else{
echo "Query Problem";
    }
}
catch(PDOException $e){
echo $e->getMessage();
}
}>
```

Видалення запису з бази.

```
<?php
include_once 'dbconfig.php';
if($_POST['del_id'])
{ $id = $_POST['del_id'];
$stmt=$db_con->prepare("DELETE FROM servise WHERE serviceID=:id");
$stmt->execute(array(':id'=>$id));
}>
```

## Додаток F

## Лістинг коду панелі адміністратора

## Файл create.php

```

<?php
require_once 'dbconfig.php';
    if($_POST)
    {
        $sv_name = $_POST['sv_name'];
        $sv_description = $_POST['sv_description'];
        $sv_price = $_POST['sv_price'];
        try{
            $stmt = $db_con->prepare("INSERT INTO service(sv_name,sv_description,sv_price)
VALUES(:ename, :edescription, :eprice)");
            $stmt->bindParam(":ename", $sv_name);
            $stmt->bindParam(":edescription", $sv_description);
            $stmt->bindParam(":eprice", $sv_price);
            if($stmt->execute())
            {echo "Successfully Added";
            }
            else{echo "Query Problem";
            }
        }
        catch(PDOException $e){
            echo $e->getMessage();
        }
    }?>

```

## Файл update.php

```

<?php
require_once 'dbconfig.php';
    if($_POST)
    {
        $id = $_POST['id'];
        $emp_name = $_POST['sv_name'];
        $emp_dept = $_POST['sv_description'];
        $emp_salary = $_POST['sv_price'];
        $stmt = $db_con->prepare("UPDATE service SET sv_name=:en, sv_description=:ed, sv_price=:es
WHERE serviceID=:id");
        $stmt->bindParam(":en", $sv_name);
        $stmt->bindParam(":ed", $sv_description); $stmt->bindParam(":es", $sv_price);
        $stmt->bindParam(":id", $serviceID);
        if($stmt->execute())
        {echo "Successfully updated";}
    }

```



```
else{ echo "Query Problem";} }
```

## Додаток G

### ЛІСТИНГ КОДУ ФУНКЦІЇ З ЗВОРОТНЬОГО ЗВ'ЯЗКУ

#### Файл cont\_us.php

```
<?php
// Check for empty fields
if(empty($_POST['name']) ||
    empty($_POST['email']) ||
    empty($_POST['phone']) ||
    empty($_POST['message']) ||
    !filter_var($_POST['email'],FILTER_VALIDATE_EMAIL))
{
    echo "No arguments Provided!";
    return false;
}

$name = strip_tags(htmlspecialchars($_POST['name']));
$email_address = strip_tags(htmlspecialchars($_POST['email']));
$phone = strip_tags(htmlspecialchars($_POST['phone']));
$message = strip_tags(htmlspecialchars($_POST['message']));

// Create the email and send the message
$to = 'Fitness@gmail.com'; // my email address
$email_subject = "Website Contact Form: $name";
$email_body = "You have received a new message from your website contact form.\n\n"."Here are the
details:\n\nName: $name\n\nEmail: $email_address\n\nPhone: $phone\n\nMessage:\n$message";
$headers = "Від: fitness.com\n";
$headers .= "Reply-To: $email_address";
mail($to,$email_subject,$email_body,$headers);
return true;
?>
```

## Додаток Н

## Лістинг коду запису на послугу

Service\_order.php

```

<?php
// Check for empty fields
if(empty($_POST['name'])  ||
    empty($_POST['email'])  ||
    empty($_POST['surname'])  ||
    empty($_POST['service'])  ||
    empty($_POST['department']))
    !filter_var($_POST['email'],FILTER_VALIDATE_EMAIL))
{
    echo "No arguments Provided!";
    return false;
}

$name = strip_tags(htmlspecialchars($_POST['name']));
$surname = strip_tags(htmlspecialchars($_POST['name']));
$email_address = strip_tags(htmlspecialchars($_POST['email']));
$phone = strip_tags(htmlspecialchars($_POST['phone']));
$service = strip_tags(htmlspecialchars($_POST['service']));
$department = strip_tags(htmlspecialchars($_POST['department']));

$to = $_POST['email'];
$email_subject = "$name";

$email_body = "Ви записалися на послугу.\n\n"."Більш детально:\n\nName: $name\n\nEmail:
$email_address\n\nPhone: $phone\n\nservice: $service
\n\ndepartment $department:";

$headers = "Від Fitness: fitness@fitness.com\n";
$headers .= "Reply-To: $email_address";

mail($to,$email_subject,$email_body,$headers);

return true;

?>

```