

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

«На правах рукопису»
УДК 004.043

«До захисту допущено»

Завідувач кафедри
І.Р. Пархомей
(підпис)

“ ” _____ 2019 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 126 «Інформаційні системи та технології»

на тему: Автоматизована система обслуговування відвідувачів
тренажерного залу

Виконав: студент шостого курсу, групи ІК-381мп
(шифр групи)

Пилипчук Олександр Станіславович _____ (підпис)
(прізвище, ім'я, по батькові)

Науковий керівник доцент, к.т.н., доцент Лісовиченко О.І. _____ (підпис)
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

Консультант _____ (підпис)
(назва розділу) (науковий ступінь, вчене звання, , прізвище, ініціали)

Рецензент к.т.н., доцент каф. АСОІУ Муха І.П. _____ (підпис)
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____ (підпис)

Київ – 2019 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

Рівень вищої освіти – другий (магістерський)

Спеціальність 126 «Інформаційні системи та технології»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ І.Р. Пархомей
(підпис)

«__» _____ 2019 р.

ЗАВДАННЯ

**на магістерську дисертацію студенту
Пилипчуку Олександрю Станіславовичу**

(прізвище, ім'я, по батькові)

1. Тема дисертації «Автоматизована система обслуговування відвідувачів тренажерного залу» _____ ,
науковий керівник дисертації доцент, к.т.н., доцент Лісовиченко О.І. _____ ,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету « 05 » 11 _____ 2019 р. № 3836-с
2. Термін подання студентом дисертації 18.11.2019 р. _____
3. Об'єкт дослідження процес оптимізації якості, витрат та часу обслуговування клієнтів _____
4. Предмет дослідження програмний комплекс «IntensFit» _____
5. Перелік завдань, які потрібно розробити аналіз предметної області і постановка задачі; об'єкт та предмет дослідження; постановка задачі; огляд існуючих рішень; аналіз інформаційного забезпечення; загальний архітектурний підхід; структура системи; технологія проектування веб-сервісу та її складова; система керування базами даних; хмарне середовище Azure; Адміністративний модуль; технологія

Хатагін; обмін даними в системі; розробка алгоритмічного та програмного забезпечення; архітектура програмного забезпечення; розробка програмного комплексу; профіль клієнта; профіль тренера; ролі в системі; прикріплення файлів; абонемент; діалог; Broadcast діалог; мета; заміри клієнта та досягнення мети; дисципліна; тренування; шаблони тренувань; оцінка; відправка ваги; відправка прийому їжі; прикріплення фото до і після; завдання; перелік блюд; push повідомлення; локалізація; конструктор звітів; структура бази даних; вимоги для роботи програмного комплексу; керівництво користувача; загальна інформація; маркетинговий аналіз стартап-проекту.

6. Орієнтовний перелік ілюстративного матеріалу схема архітектури; структура бази даних роботи з клієнтом; діаграма класів веб-сервісу; діаграма варіантів використання системи користувачами; використання системи тренером; використання системи клієнтом. _____

7. Орієнтовний перелік публікацій _____

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз предметної області	13.09.2019 р.	
2	Постановка задачі	15.09.2019 р.	
3	Аналіз інформаційного забезпечення	20.09.2019 р.	
5	Аналіз алгоритмічного забезпечення	25.09.2019 р.	
6	Розробка алгоритмічного забезпечення	15.10.2019 р.	
7	Розробка програмного забезпечення	01.11.2019 р.	
8	Маркетинговий аналіз стартап-проекту	10.11.2019 р.	
9	Висновки	15.11.2019 р.	

Студент _____
(підпис)

О. С. Пилипчук
(ініціали, прізвище)

Науковий керівник дисертації _____
(підпис)

О. І. Лісовиченко
(ініціали, прізвище)

Анотація

Об'єктом розробки є автоматизована система обслуговування відвідувачів тренажерного залу.

Мета розробки – поліпшення якості, швидкості та зручності обслуговування клієнтів в сфері послуг.

Робота присвячена розробці системи автоматизації процесу обслуговування відвідувачів тренажерного залу. У дипломі розглянуті сучасні підходи до вирішення поставленого завдання. На їх основі розроблено архітектуру системи. Система допомагає покращити обслуговування клієнтів в сфері послуг. Розроблена система складається з окремих компонентів, що робить її легко керованою та дає можливість інтеграції з іншими системами. Система побудована з використанням сучасної елементної бази і здатна вирішувати питання оптимізації роботи в сфері послуг.

Отримані результати можуть бути корисними при автоматизації аналогічних чи подібних об'єктів.

Ключові слова: автоматизована система, адміністративний модуль, мобільний додаток, програмний комплекс, управління, оптимізація, сфера послуг.

Робота містить 96 аркушів, 42 рисунків, 26 таблиць, посилання на 18 джерел, 6 демонстраційних плакатів.

Summary

The object of development is an automated system of service for the visitors of the gym.

The goal of the development is to improve the quality, speed and convenience of customer service in the field.

The work is devoted to the development of automated system for gym visitors. The diploma examines modern approaches to solving the problem. The architecture of the system is developed based on these approaches. The system helps to improve customer service in the service sector. The developed system consists of separate components, which makes it easy to manage and allows integration with other systems. The system is based on the use of a modern element base and is capable of solving the issues of optimization of the work in the service sector.

The results obtained can be useful in automating similar objects.

Keywords: automated system, the administrative module, mobile application, software package, management, optimization, service sphere.

The work contains 96 pages, 42 pictures, 26 tables, links to 18 sources, 6 demo posters.

**Пояснювальна записка
до магістерської дисертації**

на тему: *Автоматизована система обслуговування відвідувачів
тренажерного залу*

Київ – 2019 року

Перелік умовних позначень

БД – база даних;

API (application programming interface) – програмний інтерфейс;

Транзакція (transaction) — група послідовних операцій з базою даних, яка є логічною одиницею роботи з даними;

JSON (JavaScript Object Notation) – текстовий формат обміну даними;

IIS (Internet Information Services) – програмне забезпечення для розгортання веб-сервісів (входить до складу Windows);

IT – інформаційні технології.

Зміст

Перелік умовних позначень.....	7
Вступ.....	11
1 Аналіз предметної області і постановка задачі	12
1.1 Об’єкт та предмет дослідження.....	12
1.2 Постановка задачі.....	13
Висновки по розділу	15
2 Огляд існуючих рішень.....	16
2.1 Програмний комплекс «Отмечалка»	16
2.2 Програмний комплекс «Fitness Pro»	19
Висновки по розділу	22
3 Аналіз інформаційного забезпечення	23
3.1 Загальний архітектурний підхід	23
3.2 Структура системи	24
3.3 Технологія проектування веб-сервісу та її складова.....	25
3.4 Технології Representational State Transfer та Simple Object Access Protocol ...	27
3.5 Система керування базами даних.....	28
3.6 Хмарне середовище Azure	31
3.7 Адміністративний модуль.....	31
3.8 Технологія Xamarin	32
3.9 Обмін даними в системі	35
Висновки по розділу	37
4 Розробка алгоритмічного та програмного забезпечення	38

4.1 Архітектура програмного забезпечення	38
4.2 Розробка програмного комплексу	41
4.3 Профіль клієнта	43
4.4 Профіль тренера	47
4.5 Ролі в системі.....	48
4.6 Прикріплення файлів	50
4.7 Абонемент	51
4.8 Діалог	53
4.9 Broadcast діалог	55
4.10 Мета	56
4.11 Заміри клієнта та досягнення мети.....	57
4.12 Дисципліна	60
4.13 Тренування	60
4.14 Шаблони тренувань.....	65
4.15 Оцінка	66
4.16 Відправка ваги	66
4.17 Відправка прийому їжі.....	67
4.18 Прикріплення фото до і після	68
4.19 Завдання.....	68
4.20 Перелік блюд	71
4.21 Push повідомлення.....	72
4.22 Локалізація.....	73
4.23 Конструктор звітів.....	74

	10
4.24 Структура бази даних.....	74
Висновки по розділу	76
5 Вимоги для роботи програмного комплексу	77
5.1 Вимоги до технічного забезпечення	77
5.2 Вимоги до програмного забезпечення	77
Висновки по розділу	77
6 Керівництво користувача.....	78
6.1 Тренер	80
6.2 Клієнт	81
6.3 Загальна інформація	82
Висновки по розділу	82
7 Маркетинговий аналіз стартап-проекту.....	83
7.1 Опис ідеї проекту	83
7.2 Технологічний аудит ідеї проекту.....	84
7.3 Аналіз ринкових можливостей запуску стартап-проектку.....	85
7.4 Розроблення ринкової стратегії проекту.....	92
7.5 Розроблення маркетингової програми стартап-проекту	94
Висновки по розділу	96
Висновки.....	97
Перелік посилань	99
Додаток А – Демонстраційні плакати.....	101
Додаток Б – Результат перевірки роботи на співпадіння	102

Вступ

Кількість різних сервісів постійно зростає, кожного дня різні компанії пропонують багато нових послуг, які можуть вирішувати питання буденності та трати часу людиною на виконання тієї чи іншої справи. По мірі того, як стрімко розвивається Інтернет і, в загальному, розробка програмного забезпечення, керування сервісами та наданням різних послуг стає простим і надає змогу якісніше і ефективніше обслуговувати клієнтів. Додатки включають особистий кабінет, в якому користувач може самостійно виконувати різні дії, такі як: налаштування та підключення персональних даних, збереження власної інформації, можливість легкого і зручного обміну інформації з користувачами системи, перегляд результатів виконання певних дій та статистики. Завдяки розробленому функціоналу клієнти додатку мають можливість швидко отримувати зворотній зв'язок про виконання ними певних дій, завдань тощо.

Проектований програмний комплекс вирішує проблему покращення швидкості, якості, а також зручності обслуговування користувачів сфери послуг.

IntensFit – це система, яка допомагає організувати роботу з клієнтами при наданні послуг, введення обліку та контролю клієнтів в тренажерному залі, а також мобільний додаток для полегшення спілкування і співпрацювання клієнта з тренером для досягнення поставлених цілей.

1 Аналіз предметної області і постановка задачі

1.1 Об'єкт та предмет дослідження

В наш час є дуже популярним заняття спортом і ця популярність з кожним днем зростає. В зв'язку з цим відкриваються все більше тренажерних залів, які обслуговують велику кількість охочих. Та систем, що змогли би допомогти якісніше та швидше надавати послуги по обслуговуванню клієнтів не розроблено.

Програмний комплекс «IntensFit», розроблений для спортивного клубу, який надає Клієнтам послуги персональних тренувань, а також підготовки програм харчування та подальший контроль виконання завдань і досягнення результату Клієнтом з боку Тренера клубу з метою досягнення бажаного фітнес результату.

Завдання, які вирішує адміністративний модуль:

- контроль активності учасників системи та її процесів;
- параметризація сутностей системи;
- звітність;
- комунікація між користувачами;
- постановка цілей та виконання завдань;
- рольовий доступ;
- локалізація (доступно на декількох мовах);
- менеджмент абонементів та кількості занять;
- автоматичне регулювання цілей та завдань для виконання клієнтами;
- інше.

Програма для схуднення «Інтенсив» - це основна програма для тренажерного клубу, який буде використовувати дану систему. Комплекс дозволяє налаштовувати та конфігурувати програму різними інструментами (кількість тренувань, кількість занять в одному абонементі, призупинення дії абонементу на деякий час, призначення завдань як індивідуальних, так і завдань, що потрібно виконати безпосередньо в тренажерному клубі).

Програма «Інтенсив» включає в себе:

- певну кількість відвідувань тренажерного залу за період абонементу;
- індивідуальні тренування в тренажерному залі;
- індивідуальну програму харчування на кожен день;
- контролювання досягнення результатів та їх коригування (якщо потрібно).

Основна мета розробки програмного комплексу – це автоматизація різних процесів комунікації з клієнтами, поставлення і регулювання досягнення цілей клієнтів та оптимізація витрати часу на ці процеси.

Предмет досліджень в даній роботі – це програмний комплекс «IntensFit».

Об'єкт досліджень – це оптимізація якості та витрат часу на обслуговування клієнтів.

1.2 Постановка задачі

Метою роботи є розробка програмного забезпечення на комп'ютер для платформи Windows та мобільного додатку для платформи Android і IOS. Задля досягнення поставленої мети необхідно вирішити наступні задачі:

- провести аналіз функціональності існуючих подібних рішень та виділити функції, які потрібно включити в розробку своєї системи, а які розробити;
- розробити архітектуру системи, вибрати технології для розробки окремих частин системи;
- розробити програмний комплекс, який складається з додатку для платформи Windows, мобільний додаток для платформ Android і IOS, а також Web API для взаємодії їх з БД.

Створена система повинна відповідати наступним вимогам:

- реєстрація користувачів;

- призначення користувачеві програму тренування (абонемент);
- створення діалогу для спілкування тренера та клієнта;
- створення діалогу з більше, ніж двома учасниками (груповий діалог);
- налаштування цілей та завдань для клієнта;
- виконання різних завдань клієнтом та тренером;
- онлайн відображення результатів виконання завдань;
- налаштування відправлення нагадувань клієнтам про необхідність

виконати поставлене завдання;

- розподілення прав для користувачів системи;
- локалізація системи;
- влаштований конструктор звітів;
- мобільний додаток з обов'язковою авторизацією і автентифікацією;
- комунікація учасників системи;
- прикріплення фотокартки клієнта до і після початку занять з

тренером;

- завчасно підготовлений набір блюд для харчування;
- створення, редагування тренувань та одразу, або через деякий час,

призначення їх клієнтові;

- внесення в систему певних замірів тіла клієнта для встановлення бажаних цілей і визначення результатів.

- інтуїтивно зрозумілий інтерфейс;

- простота у використанні.

У розділі проаналізовано проблематику в сфері надання послуг клієнтам, предметної області та сформульовано основні вимоги до програмного комплексу. Визначено функціонал, якого не вистачає для повноцінного продукту. В результаті проведеного аналізу сформульована постановка задачі.

Розроблюваний програмний комплекс «IntensFit» призначений для спортивного клубу, який надає послуги персональних тренувань, зручний додаток

як для клієнтів, так і для тренерів в якому є можливість спілкуватися, створювати та виконувати тренування, переглядати статистику, виконувати завдання тощо.

Висновки по розділу

У першому розділі розглянуто об'єкт та предмет дослідження – автоматизація процесу надання послуг клієнтам, якість обслуговування та зменшення витрат часу на це обслуговування. Після чого сформульовано цільові вимоги до розроблюваного програмного комплексу. Для збільшення конкурентноспроможності, розробки сильних сторін та якості продукту проведемо аналіз існуючих рішень, щоб підтвердити актуальність сформульованих вимог до продукту.

2 Огляд існуючих рішень

При пошуку подібних розробок для рішення поставлених задач було знайдено декілька варіантів систем, які вирішують ту чи іншу їх частину. Але не в повному обсязі, як цього хотілось би.

2.1 Програмний комплекс «Отмечалка»

Отмечалка – система, яка створена для автоматизації студій, дитячих центрів, мовних і музичних шкіл та фітнес клубів. Є простим, зручним та доступним інструментом ведення бізнесу для цього напрямку.

З системою Отмечалка ви отримаєте такі переваги:

- робота з клієнтами (управління, додавання медичних довідок, видача абонементів);
- облік відвідуваності (рисунок 1);

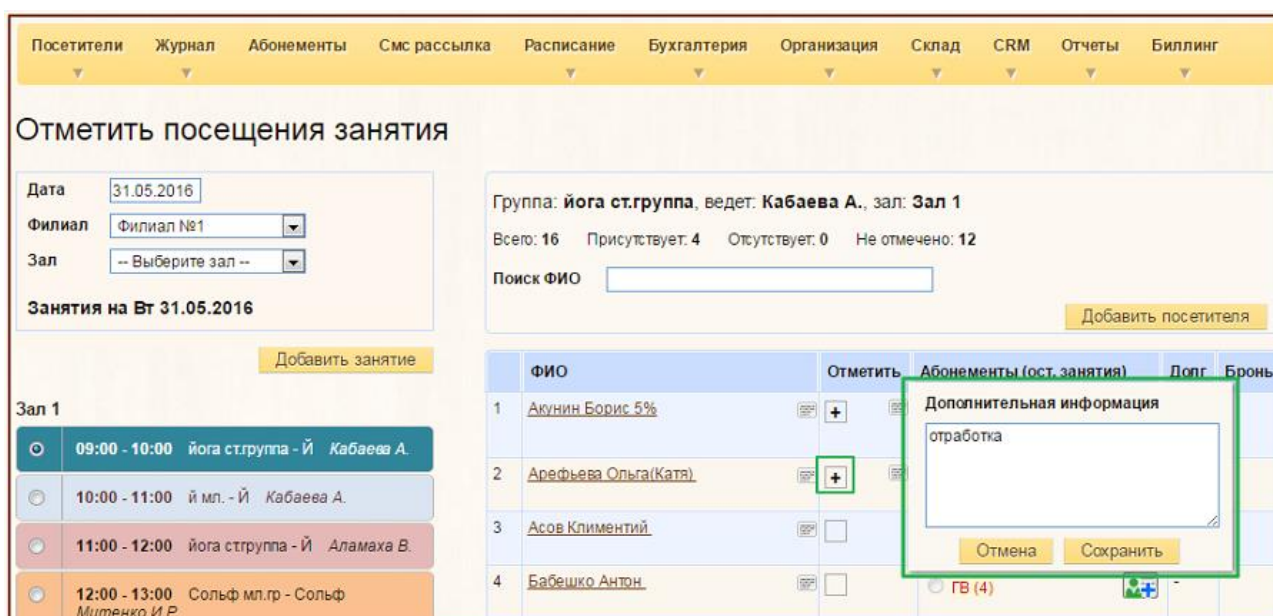


Рисунок 1 – Облік відвідуваності [13]

- розподілення ролей доступу до системи;
- облік боргів клієнтів;
- модуль «Депозит клієнтів». З депозитного рахунку клієнт має можливість оплатити абонемент повністю або частково чи провести розрахунок за будь-який інший товар клубу;
- модуль «Особистий кабінет користувача»;
- робота з розкладом занять, що дозволяє бронювати місця на різних заняттях та можливістю одразу декількох броней і вказівки коментарів до них
- адміністративний модуль системи дозволяє вести комунікацію з клієнтами чи іншу взаємодію з ними (рисунок 2);

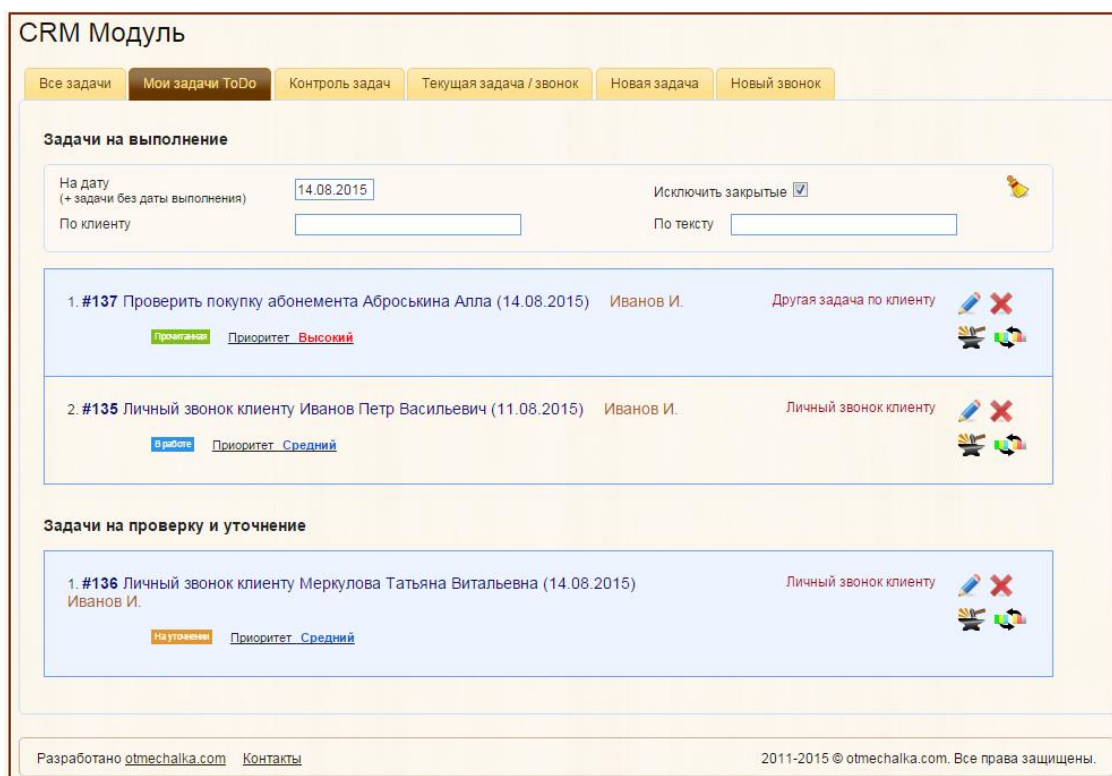


Рисунок 2 – Вигляд CRM модуля [13]

- автоматичні дії (відправка повідомлень при закінченні строку дії абонементу, автоматична зміна статусу клієнта);

- мобільний додаток під платформу Android (рисунок 3).

Наявний мобільний додаток, який дозволяє:

- більш швидше та зручніше виконувати пошук клієнта, відмічати відвідування співробітників;
- зменшити витрати клубу на покупку сканерів за рахунок сканування штрих-кодів, які знаходяться на клубних картах, використовуючи камеру смартфона.

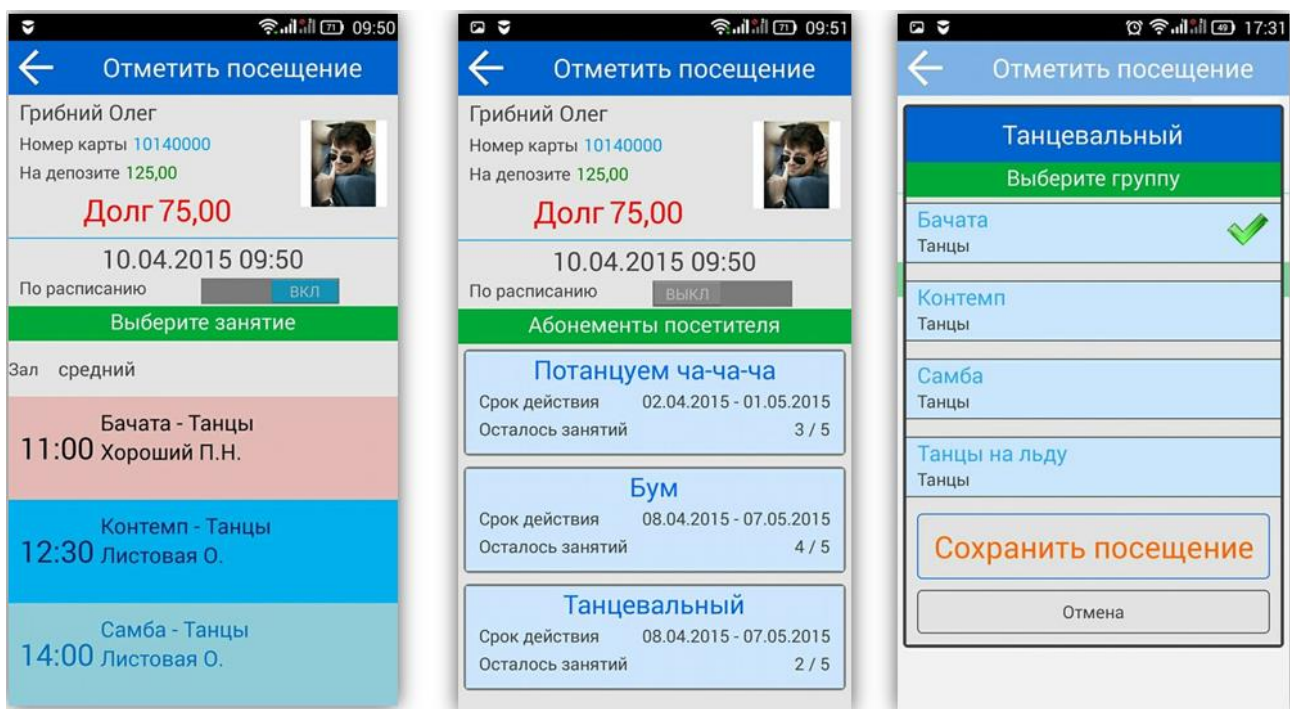


Рисунок 3 – Мобільний додаток [18]

Клієнти мають можливість переглянути наступну інформацію:

- активні і архівовані абонементи, відвідування кількість занять, які залишилися по кожному абонементу;
- список платежів за період часу;
- стан та операції по депозитному рахунку;
- відвідування за певний період;

- розклад занять;
- свій профіль (ПІБ, номер телефону та e-mail).

Зручність керування своєю справою та бачення повної картини досягається завдяки зосередженому в одному місці функціоналу та можливості маніпулювання різними функціями системи швидко. Всі клієнти матимуть свій особистий кабінет, в якому вони бачать потрібну інформацію щодо своїх замовлень, звернень до служби підтримки, звіти, інші документи і власні відгуки.

2.2 Програмний комплекс «Fitness Pro»

Fitness Pro – інструмент, який допоможе вирішити будь-які питання бізнесу, а саме: фітнес клубів, спортивних та тренажерних залів. Дана програма надає можливість вести базу клієнтів, продаж абонементів, контроль і відмітки про відвідування, записи на групові заняття і багато іншого. Існує можливість використовувати програмний комплекс для мережі спортивних клубів.

В системі наявні наступні можливості:

- додавання клієнтів та перегляд їх профілів, що розподілені на категорії. Цей функціонал надає широкий спектр можливостей – починаючи від додавання та редагування анкет до зняття або внесення коштів з/на депозитний рахунок і перегляд покупок (рисунок 4);
- перегляд розкладу з застосуванням різноманітних фільтрів (наприклад, по тренерам, групам, клубам чи залам тощо), отримувати доступ до контактів співробітників та записуватися на заняття;
- контролювати бізнес в режимі 24/7;
- організувати роботу відділу продажу;
- підключити систему контролю доступу;

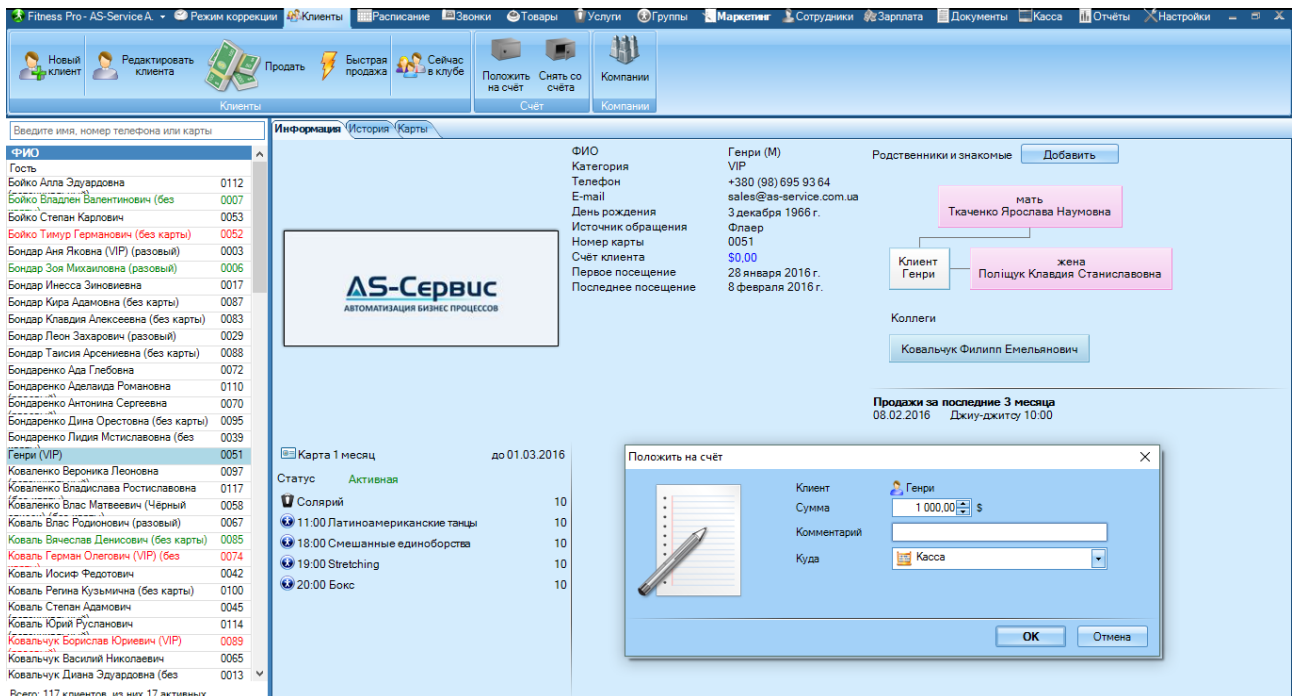


Рисунок 4 – Сторінка «Клієнти» [18]

- введення каталогу та планування групових занять;
- відображення списку послуг (рисунок 5);
- систематизація роботи з абонементом і картами;
- керування обігом товарів;
- для розвантаження адміністратора створено робоче місце касира, яке дозволить виконувати всі операції по розрахунках клієнтів;
- збереження документів (договори с клієнтами, співробітниками, інструкції тощо);
- спростити процес нарахування заробітної платні тренерам і персоналу;
- скоротити витрати на неефективну рекламу;
- отримувати аналітику по акціям, які діють;
- вітати клієнтів з днем народження або нагадувати про закінчення строку дії картки;
- відправляти SMS і Email розсилку безпосередньо з програми;

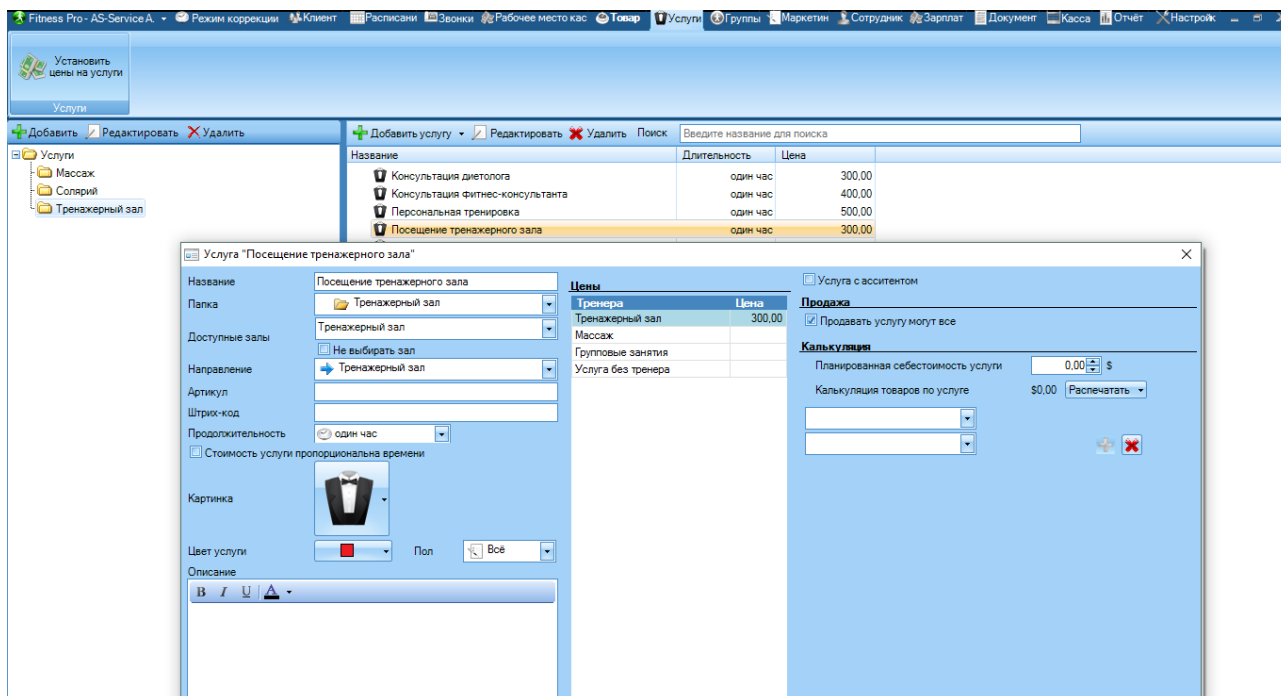


Рисунок 5 – Сторінка «Послуги» [18]

– сторінка звітів і налаштувань (рисунок 6).

Зручність користування – це перший пріоритет системи. Розробниками було створено систему, яку можна максимально налаштувати під потреби кожного. Вона має широкий набір функціональних можливостей.

Fitness Pro допоможе організувати ефективний робочий процес, поліпшити якість обслуговування клієнтів, автоматизувати виконання рутинних завдань і економії часу, а також виключення помилки людського фактору. Користувач отримає інструмент для аналітики ефективності роботи бізнесу на всіх рівнях, що допоможе досягти поставлених цілей.

Нааявна можливість замовити безкоштовну онлайн презентацію програми, за допомогою якої може бути продемонстровано покупцеві як програма для фітнес клубів і тренажерних залів допомагає користувачам виконувати різні завдання, ефективно керувати закладом та збільшувати прибуток.

Підтримка. Фахівці служби підтримки завжди на зв'язку для комфортної роботи. Вони допоможуть у вирішенні будь-яких завдань і оперативно проконсультують по питанням, які будуть виникати.

Клиент	Телефон	Менеджер	Сервис-менедж	Новый клиент	Название	Дата продажи	Начало карты	Конец карты	Стоимость карты
Ткачук Мирослав Якович	+380 (352) 89 17	нет	нет	+	Карта 3 месяца	04.02.2016	04.02.2016	04.05.2016	1 200,00
Бондаренко Ада Глебовна	+380 (4869) 2 42	нет	нет	+	Карта 1 месяц	05.02.2016	05.02.2016	05.03.2016	300,00
Марченко Виктор	+380 (39) 521 89	нет	нет	+	Карта полгода	05.02.2016	05.02.2016	05.08.2016	1 800,00
Кравченко Валентин	+380 (73) 111 36	нет	нет	+	Персональная	05.02.2016	05.02.2016	05.05.2016	2 484,00
Мельник Андрей Захарович	+380 (39) 228 36	нет	нет	+	Карта полгода	06.02.2016	06.02.2016	06.08.2016	1 800,00
Олійник Марина Леоновна	+380 (5323) 7 69	нет	нет	+	5%	06.02.2016	06.02.2016	постоянная карта	0,00
Олійник Марина Леоновна	+380 (5323) 7 69	нет	нет	+	Карта 3 месяца	11.02.2016	11.02.2016	11.05.2016	1 200,00
Олійник Марина Леоновна	+380 (5323) 7 69	нет	нет	+	Карта 3 месяца	11.02.2016	11.02.2016	11.05.2016	1 200,00
Марченко Клар	+380 (73) 410 87	нет	нет	+	Карта 1 год	07.02.2016	07.02.2016	07.02.2017	4 200,00
Ковальчук Василий	+380 (4475) 7 95	нет	нет	+	Карта полгода	08.02.2016	08.02.2016	08.08.2016	1 800,00
Руденко Назар	+380 (4841) 6 18	нет	нет	+	Бонусная карта	08.02.2016	08.02.2016	постоянная карта	1 800,00
Шевчук Тарас Владислав	+380 (67) 734 72	нет	нет	+	Карта 1 год	08.02.2016	08.02.2016	08.02.2017	4 200,00
Коваль Иосиф Федотович	+380 (352) 06 42	нет	нет	+	Карта 1 год	09.02.2016	09.02.2016	09.02.2017	4 200,00
Олійник Федот	+380 (4869) 9 88	нет	нет	+	Бонусная карта	10.02.2016	10.02.2016	постоянная карта	0,00
Руденко Арсений	+380 (4869) 9 88	нет	нет	+	Карта полгода	09.02.2016	09.02.2016	09.08.2016	1 800,00
Бойко Степан Карлович	+380 (48) 781 01	нет	нет	+	5%	10.02.2016	10.02.2016	постоянная карта	0,00
Бондар Инесса Зиновиевна	+380 (4663) 9 79	нет	нет	+	Бонусная карта	10.02.2016	10.02.2016	постоянная карта	0,00
Марченко Ивесса Зиновиевна	+380 (652) 33 14	нет	нет	+	5%	11.02.2016	11.02.2016	постоянная карта	0,00
Марченко Сергеевна	+380 (5385) 8 26	нет	нет	+	Карта 1 месяц	11.02.2016	11.02.2016	11.03.2016	300,00
Генри	+380 (98) 695 93	нет	нет	+	Карта 1 месяц	01.02.2016	01.02.2016	01.03.2016	3 000,00
Всего (20)									29 484,00

Рисунок 6 – Сторінка «Звіти» та «Налаштування» [18]

Є функціонал, який не так часто використовується. Ряд параметрів потребує одноразового внесення в систему, такі як: автоматичні нагадування, інформація про клуб, основна валюта, правила продажу співробітникам тощо. У системі наявне розподілення доступу до різної інформації системи або функцій системи. Це залежить від призначення ролей користувачам.

Висновки по розділу

Проведено аналіз існуючих рішень, їх функціональні можливості та частини, з яких складаються ці рішення. На основі цього підтверджено визначені вимоги як до системи, так і до окремих її частин. Для реалізації рішення необхідний аналіз інформаційного забезпечення та визначення архітектури.

3 Аналіз інформаційного забезпечення

3.1 Загальний архітектурний підхід

Реалізація рішення проводилася з використанням сервісно-орієнтованої архітектури.

Сервіс-орієнтована архітектура (service-oriented architecture, SOA) придумана в кінці 1980-х та означає модульний підхід до розробки програмного забезпечення. Цей підхід заснований на використанні розподілених, слабо зв'язних замінюваних компонентів (частин системи), які містять в собі стандартизовані інтерфейси для взаємодії по стандартизованим протоколам. Компоненти таких систем можуть знаходитися в різних вузлах мережі та абсолютно не залежати один від одного.

Комплекси, які спроектовані та розроблені у відповідності з такою архітектурою, зазвичай реалізуються як набір веб-служб, взаємодіючих по протоколу SOAP – протокол для обміну структурованими даними у вигляді XML. Також присутні і реалізації на основі інших стилів. Наприклад, до них відноситься архітектурний стиль REST. REST – це не протокол взаємодії, а підхід до реалізації взаємодії, який допомагає передавати стан об'єктів. Така реалізація включає в собі запит з даними, які потрібно передати сервісу або клієнту. Формат даних може бути різним, частіше за все використовують JSON або той же XML.

Основною відмінністю REST сервісу в сервісно-орієнтованій архітектурі є те, що сервіс не зберігає стану клієнта з якого отримує запити, а сприймає кожен наступний запит як новий. Для цього клієнти повинні містити дані про авторизацію та автентифікацію (якщо така наявна в сервісі) для виконання різних операцій.

Сервісно-орієнтована архітектура дає можливість гнучкого способу комбінування і багаторазового або повторного використання тих чи інших компонентів для побудови розподілених програмних комплексів. Завдяки цьому є

можливість створювати системи незалежно від платформ і технологій за допомогою яких були розроблені компоненти. Також, такий підхід допомагає простіше масштабувати систему, тестувати окремі компоненти системи, повторно використовувати різні частини системи тощо.

Важливою перевагою є зручність оновлення окремих компонентів та використання технологій контейнеризації. Контейнеризація дозволяє створювати декілька екземплярів одного і того ж сервісу і за допомогою спеціальних інструментів розподіляти запити від клієнтів на ці екземпляри. Ця перевага полегшує оновлювати окремі компоненти при великих масштабах або сильно навантажених систем.

3.2 Структура системи

Програмний комплекс включає наступні компоненти:

- адміністративний модуль – додаток для платформи Windows – це клієнтська частина. За допомогою цього компоненту відбувається керування системою;
- мобільні додатки для платформи Android та iOS – клієнтська частина;
- зовнішня бібліотека – ядро системи. Включає об'єкти та методи, за допомогою яких відбувається оперування цими об'єктами;
- ASP.NET Web API – веб-сервіс, призначений для використання десктопним і мобільним додатком. Ключова функція такого сервісу – це обмін та модифікація даних системи;
- база даних MS SQL – сховище даних.

3.3 Технологія проектування веб-сервісу та її складова

ASP.NET (Active Server Pages для .NET) – це технологія створення веб-застосунків, в склад якої входить:

- програмна інфраструктура;
- веб-сервіси;

ASP.NET входить до складу платформи Microsoft .NET Framework і є розвитком більш застарілої технології Microsoft ASP. Так як вся розробка проходить з використанням продуктів компанії Microsoft і мови програмування C#, тому обрана технологія цілком підійде для розробки веб-сервісу.

Багато зручних рішень, таких як: автентифікація, авторизація, збереження стану сторінки, швидкість у використанні та проектуванні, розподілення запитів, заміна частин системи, структури даних для взаємодії, економія трафіку (як наслідок – прискорення роботи) – все це влаштовано в ядро технології ASP.NET. Великий набір елементів (рисунок 7) управління і бібліотек класів для вирішення різноманітних завдань дає змогу швидко розробляти програмні застосунки. ASP.NET спирається на різні можливості платформи .NET – розподіл візуальної частини і бізнес-логіки.

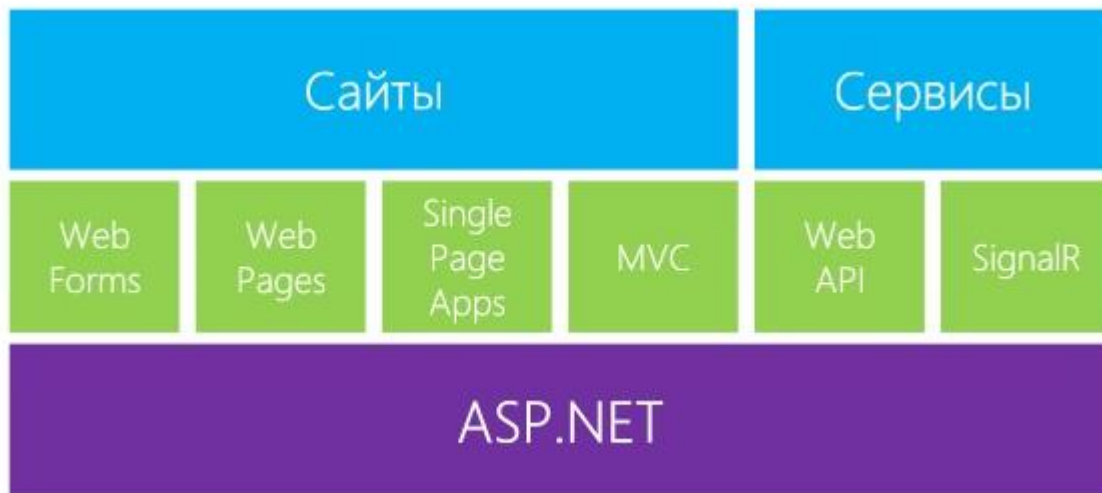


Рисунок 7 – Платформені компоненти ASP.NET [14]

ASP.NET Web API – це сервіс, що не залежно від архітектурних реалізацій клієнтів, які взаємодіють, можете однаково опрацьовувати вхідні дані та повертати результат.

У контексті Web API сервісу використовується термін контролер, який базується на тому, щоб створити просту та необхідну подію в результаті запитів від клієнтів. За рахунок того, що є популярним використанням HTTP протоколу для обміну даних, ця технологія підтримує основні принципи веб-комунікації HTTP.

Деякі переваги використання Web API:

- значно менша кількість написаного коду для реалізації функціоналу в порівнянні з SOAP веб-службою на WCF;
- більша швидкість передачі даних (завдяки можливості обирати формат обміну даними);
- контроллери цієї технології використовують стиль REST;
- зручність розповсюдження та повторне використання сервісу (рисунок 8).

Сервіс використовує підхід (стиль) REST-архітектури з такими методами HTTP як:

- PUT;
- GET;
- POST;
- DELETE.

Визначені програмістом методи контролера позначаються відповідними атрибутами та зіставляються з однойменними методами, що представлені вище.

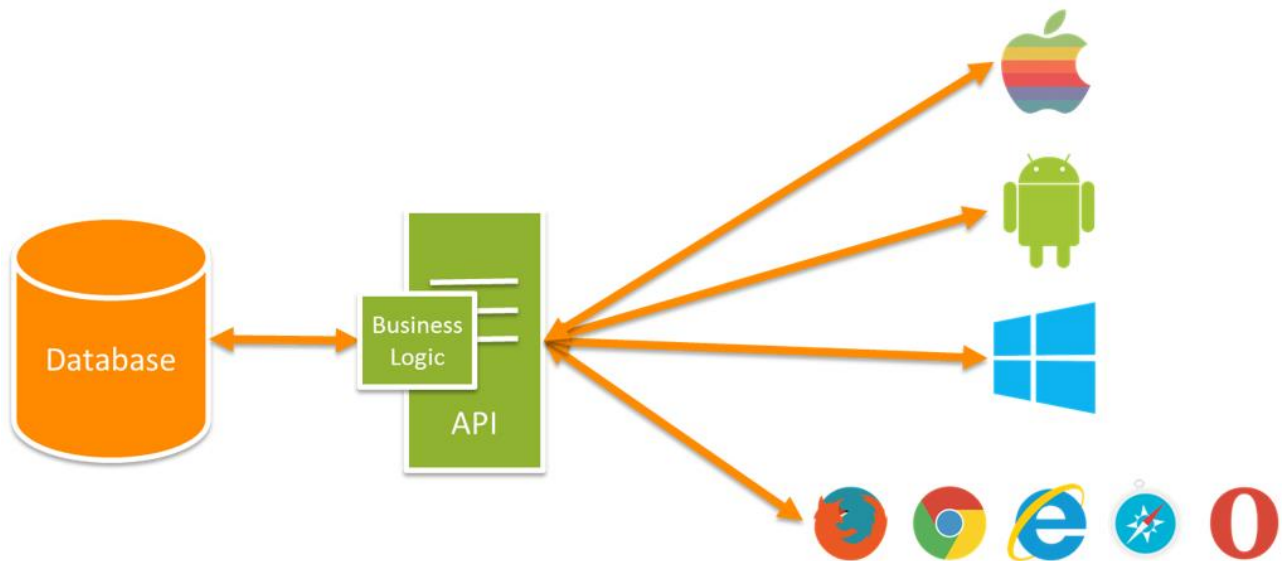


Рисунок 8 – Схематичне зображення переваги використання API [12]

3.4 Технології Representational State Transfer та Simple Object Access Protocol

REST (Representational State Transfer) – це архітектурний підхід до програмування сервісів, а SOAP (Simple Object Access Protocol) – це протоколи і стандарти.

Основною перевагою архітектури REST є простота. Простота ідеї, простота розробки та додавання додаткової функціональності до RESTful-додатків. Цей підхід простий і універсальний, що спочатку складно вловити. Тому до аналізу переваг та недоліків шляхом порівняння SOAP і REST.

- SOAP – це сімейство протоколів та різних стандартів. Звідси випливає те, що цей варіант більш великогаговий і складний варіант з точки зору машинної обробки. І як висновок – REST працює швидше.

- SOAP використовує HTTP протокол як транспортний, в той час як REST базується на ньому. Це означає, що всі існуючі розробки на базі цього протоколу, наприклад, такі як серверне кешування, масштабування, працюють в REST архітектурі, а для SOAP необхідно шукати інші засоби. З другої сторони SOAP сервіси отримують можливість працювати з будь-яким протоколом

транспортного рівня замість HTTP. Але часто від цієї можливості ніякої практичної користі.

- Відповідь від REST сервісу може бути представлена у різних форматах, а SOAP – тільки XML. І це дійсно важливий фактор.

- «REST і SOAP» можна перефразувати таким чином, як «Простота і Стандарти». Це ґрунтується на тому, що для SOAP ми маємо протокол WSDL (схема опису сервісу з усією необхідною інформацією для клієнта) для опису веб-сервісу, а з боку REST таких схем опису не потрібно.

- Один із головних аспектів – обробка помилок. У SOAP така обробка повністю стандартизована, а з REST з’являється можливість використовувати давно відомі різні коди помилок HTTP, яких існує достатньо велика кількість.

- Одна з ключових думок в порівнянні – SOAP працює з операціями, а REST – з ресурсами. Завдяки цьому факту в сукупності з відсутністю клієнтського стану у RESTful сервісів, приводить до того, що такі речі як транзакції або інша складна логіка повинна реалізовуватися при використанні SOAP.

SOAP і REST – це не конкуренти. Дані підходи представляють різні вагові категорії і навряд чи знайдеться завдання, для якої буде складно сказати, який підхід раціональніше використовувати – SOAP чи REST. Тому кожен вибирає технологію за своєю потребою та часом на її реалізацію.

Іноді притримуватися REST стилю не так просто, особливо на початку розробки, але згодом це полегшує еволюцію на стороні сервісу і стійкістю клієнтів до змін. Якщо потрібно розробити сервіс швидко і легко, REST – саме те, що потрібно.

3.5 Система керування базами даних

Для сховища даних було обрано технологію Microsoft SQL Server – це розробка компанії Microsoft, яка представляє собою систему керування

реляційними базами даних. До таких типів бази даних використовується мова запитів Transaction-SQL. Існують також і інші система керування базами даних (СКБД), такі як SQL та PL/SQL.

SQL Server – це одна з найпопулярніших систем управління базами даних. Така СУБД підходить для різних проектів – від невеликих додатків до великих високонавантажених проектів.

Характеристики SQL Server такі:

- продуктивність. SQL Server працює дуже швидко;
- безпека та надійність. SQL Server надає шифрування даних;
- простота. З такою СУБД достатньо легко працювати та вести її адміністрування.

Центральний аспект такого сервера (так як і в будь-якій іншій СУБД) – це база даних. Вона являє собою сховище даних, які організовані певним чином. Доволі часто база даних представляє файл на жорсткому диску. Для організації баз даних MS SQL Server використовує реляційну модель. Дана модель баз даних розроблена в 1970 році Едгаром Коддом і на сьогоднішній день є стандартом для організації баз даних.

Реляційна модель передбачає зберігання даних у вигляді таблиць. Кожна з таких таблиць містить рядки і стовпці. Все організовано таким чином, що кожен рядок – це окремий об'єкт, а стовпці – атрибути цього об'єкта. Для ідентифікації кожного рядка (об'єкта) в рамках таблиці використовується первинний ключ (primary key). Первинний ключ може складатися з одного або декількох стовпців. Це є унікальний ідентифікатор запису в таблиці і він може використовуватися при пошуках в запитах. Відповідно до цього два рядки не можуть мати один і той же первинний ключ. За допомогою ключів можна об'єднувати таблиці в одному запиті, що є зручним при запитах.

Для взаємодії з базою даних застосовується мова SQL (Structured Query Language). SQL – це мова програмування, яка допомагає в управлінні даними, що

зберігаються в системі керування базами даних, а також для можливості потокової обробки даних в системі управління реляційними потоками даних.

Залежно від завдання, яке виконує певна команда T-SQL, може належати до одного з наступних типів:

1. DDL (Data Definition Language, мова визначення даних). До цього типу належать команди, які дозволяють створити базу даних, таблиці, індекси, збережені процедури тощо. В загальному визначають дані. До цього типу можна віднести наступні команди:

а) CREATE: створює об'єкти бази даних (саму базу даних, таблиці, індекси і т.д.);

б) ALTER: змінює об'єкти бази даних;

в) DROP: видаляє об'єкти бази даних;

г) TRUNCATE: видаляє всі дані з таблиць;

2. DML (Data Manipulation Language, мова маніпуляції даними). До цього типу відносяться команди вибору, оновлення, додавання та видалення даних – власне ті команди, за допомогою яких можна управляти даними. До такого типу належать команди:

а) SELECT: витягує дані з БД;

б) UPDATE: оновлює дані;

в) INSERT: додає нові дані;

г) DELETE: видаляє дані;

3. DCL (Data Control Language, мова управління доступу до даних). Це команди, що керують правами доступу до даних. Зокрема, такі як:

а) GRANT: надає права для доступу до даних;

б) REVOKE: відкликає права на доступ до даних;

3.6 Хмарне середовище Azure

Веб-сервіс програмного комплексу IntensFit та сховище даних розміщуються в хмарному середовищі Azure.

Azure – хмарна платформа від компанії Microsoft. Така технологія надає можливість просто і зручно розміщувати та використовувати розроблені додатки і зберігати дані на віддалених серверах. Такі сервери можуть бути розміщені в різних точках світу в спеціалізованих приміщеннях, які називаються дата-центрами. Microsoft Azure – це постійно розширюваний набір хмарних служб, які допомагають організації вирішення різних бізнес-завдань. Технологія допомагає легко розгортати додатки та управляти ними з використанням зручних для кожного інструментів і платформ.

Для кожного сервісу, бази даних тощо, запускається віртуальна машина до якої є доступ кожного моменту і це є зручним для підключення будь-яких клієнтів. Також наявна авторизація та автентифікація, налаштування доступу до своїх екземплярів, які розміщені в такому хмарному середовищі. Є можливість обмеження по кількості запитів, частоті їх та розмірах повідомлень тощо.

3.7 Адміністративний модуль

Адміністративний модуль, перша клієнтської частини по відношенню до веб-сервісу, написаний з використанням технології DevExpress. DevExpress Winforms – це програмний пакет, до складу якого входить велика кількість елементів керування і бібліотек, що розширюють наявний функціонал Windows Forms. Використання такої технології дозволяє швидко розробляти гнучкі налаштовувані додатки. Особливо це підходить для конкретних бізнес задач, де потрібно отримати швидкий результат за мінімум витраченого часу.

3.8 Технологія Xamarin

Мобільні додатки для платформи Android і iOS, наступні клієнти веб-сервісу, створені з використанням технології Xamarin Forms.

Xamarin – являє собою платформу, яка націлена на створення кросплатформених додатків для Android, iOS і Windows Phone.

Xamarin дозволяє створювати єдину бізнес-логіку додатка з використанням мови C# і платформи .NET відразу для всіх трьох платформ – Android, iOS та UWP (Windows Phone) (рисунок 9).

Деякі переваги використання Xamarin:

- при розробці додатку створюється єдиний код для всіх платформ;
- Xamarin надає прямий доступ до нативних API кожної з платформ, що дозволяє взаємодіяти з ядром мобільної платформи та виконувати багато налаштувань;
- створюючи додатків є можливість використовувати платформу .NET і мова програмування C# (а також F#), які досить продуктивні та водночас ясні і прості для освоєння і застосування (особливо для тих, хто працює з даною платформою і не має справи з мобільною розробкою);
- Xamarin підтримує декілька платформ, основними з яких є Android, iOS, UWP, Tizen, а також додаткові – MacOS, WPF, GTK# (плануються в майбутньому).

Програмний проект такого типу містить три основні частини:

- PCL (portable class library) – бібліотека коду, яка містить основну бізнес-логіку мобільного додатку;
- проект Android;
- проект iOS.

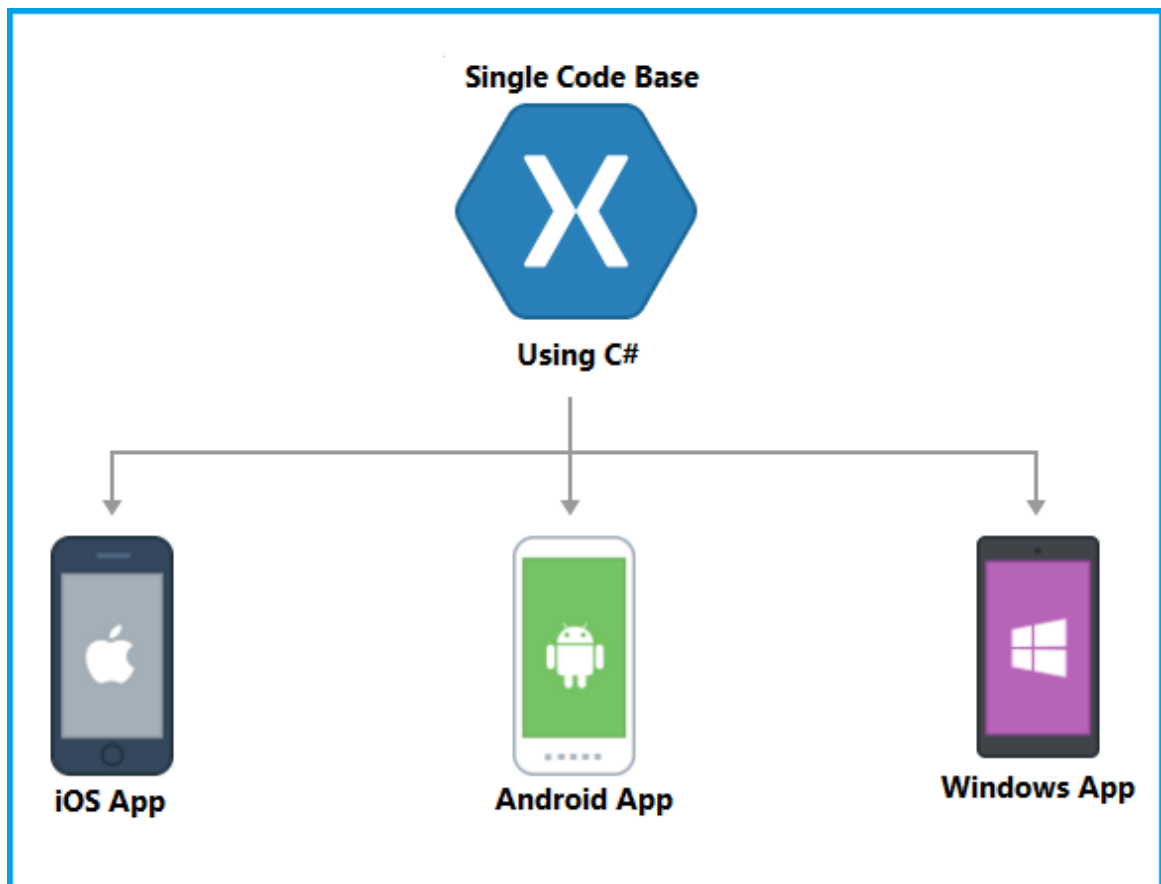


Рисунок 9 – Повторне використання коду завдяки технології Xamarin [12]

Бізнес-логіка написана з використанням підходу програмування Model-View-ViewModel (MVVM) (рисунок 10). Цей підхід ґрунтується на розділенні функціональної частини програми на три ключові:

- View – це представлення або призначений для користувача інтерфейс. Наприклад, форма з таблицею для відображення даних або форма перегляду чи редагування анкети користувача тощо;
- Model – це модель або дані, що використовуються в додатку. Кожна модель має свій набір властивостей з типом даних для кожного;
- ViewModel – це є проміжним шаром між представленням і даними, який забезпечує їх взаємодію. За допомогою цього шару надається можливість для модифікації, збагачення або видозміни даних до такого виду, який потрібен

кінцевому користувачу на екрані (додаткова обробка даних для приведення в більш зрозумілий або правильний користувачеві вигляд).

Перевагою використання даного підходу програмування є менша зв'язаність між компонентами додатку та розподілення відповідальності між ними, можливість розробки додатку декількома розробниками, які працюють над різними компонентами.



Рисунок 10 – Модель MVVM [15]

Model відповідає за дані, View відповідає за графічний інтерфейс, а ViewModel – за бізнес-логіку програми. Легкість реалізації обраного підходу в Xamarin стало можливим завдяки раніше розглянутому механізму прив'язки.

Шаблон MVVM допомагає в створенні додатків, в яких різні аспекти, такі як логіка введення, бізнес-логіка чи логіка інтерфейсу, розподілені і в той же час тісно взаємодіють один з одним. Описана вище схема вказує на конкретне розподілення написання коду в додатку. Навігація по формам додатку відбувається на основі стеку організованого за так званим принципом FIFO (first input-first output) зображеного на рисунку 11.

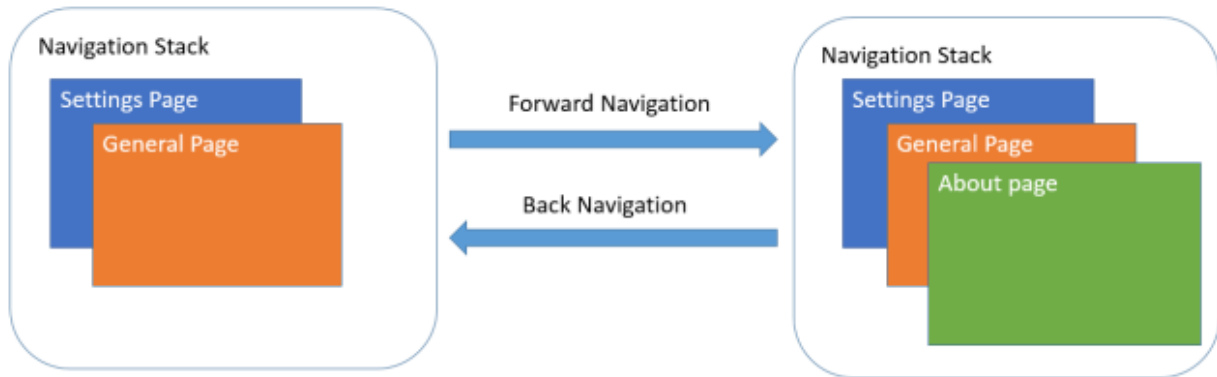


Рисунок 11 – Навігація в додатку розробленому на Xamarin Forms [16]

Xamarin MVVM має такі переваги:

- повторне використання коду понад 90%. Кросплатформена розробка мобільних додатків. Це значить, що при написанні коду відбувається одразу для двох мобільних платформ;
- влаштовані Xamarin Forms макети;
- можливість швидко і зручно будувати дизайн додатку;
- структуроване та зрозуміле керування великим та складним додатком, використовуючи поділ на модель, представлення і представлення моделі.

3.9 Обмін даними в системі

Центральним генератором обміну даними в системі виступає веб-сервіс. Для цієї можливості сервіс інтегровано з зовнішніми сервісами, такими як SignalR, що в більшості випадків використовуються для оповіщення клієнтів про нові повідомлення в діалозі та з сервісом PUSH повідомлень, який носить назву Firebase Cloud Messaging (FCM).

SignalR – це бібліотека, яка спрощує процес додавання деяких, завчасно визначених веб-функцій в реальному часі для додатків. Веб-функціональність в реальному часі - це можливість миттєвої передачі визначеного коду (розуміється

виконання певних дій визначених бізнес-логікою) підключеним клієнтам по мірі доступності. Така можливість не потребує чекати запиту від про нові дані, а відправляти автоматичне так зване повідомлення про це.

Для конфігурації на сервері потребується реалізація двох видів API:

- PersistentConnection API – це низькорівневий;
- Hub API – високорівневий.

В даному випадку Hub спирається на PersistentConnection, що зображено на рисунку 12. Можливе використання будь-якого з вище описаних та в більшості випадків достатньо можливостей реалізованих Hub API.

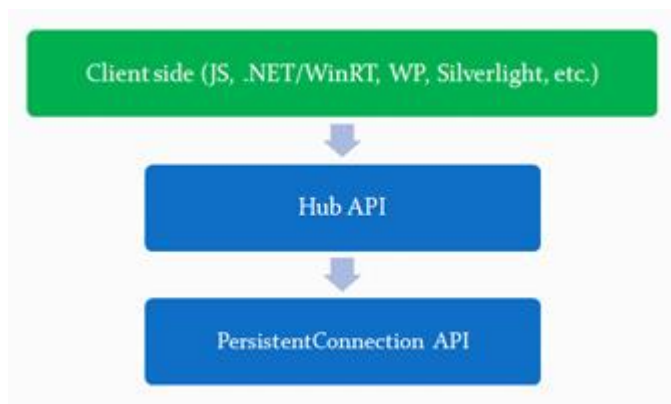


Рисунок 12 – Взаємодії SignalR з підключеними клієнтами [17]

Firebase cloud messaging (FCM) – це безкоштовний сервіс для відправки повідомлень з серверів та сервісів на обрані та налаштовані додатки пристроїв iOS, Android і Chrome.

Сервіс дозволяє відправляти повідомлення як на окремі пристрої, так і на групу пристроїв (рисунок 13), а також на пристрої, де оформлена підписка на конкретну тему. FCM може налаштовуватися для відправлення повідомлення серверу, а також на пристрої, що входять в групу.



Рисунок 13 – Зображення принципу роботи FCM [12]

Для автоматичного виконання певних дій по визначеному розкладу використано відкриту бібліотеку Quartz Scheduler (планувальник). Бібліотеку налаштовано і додано до веб-сервісу. Для роботи планувальника потрібно визначити наступні кроки:

- дію, яку потрібно запустити або виконати виконувати;
- і подію (тригер), що почне виконання дії або вказати дату та час виконання цієї дії.

Висновки по розділу

В розділі розглянуто та проаналізовано способи та технології для створення окремих компонентів програмного комплексу, наведений опис, недоліки та характеристики обраних технологій для реалізації рішення. На основі цього відбувається розробка алгоритмічного та програмного забезпечення системи.

4 Розробка алгоритмічного та програмного забезпечення

4.1 Архітектура програмного забезпечення

Програмний комплекс «IntensFit» розроблено відповідно до рисунка 14.

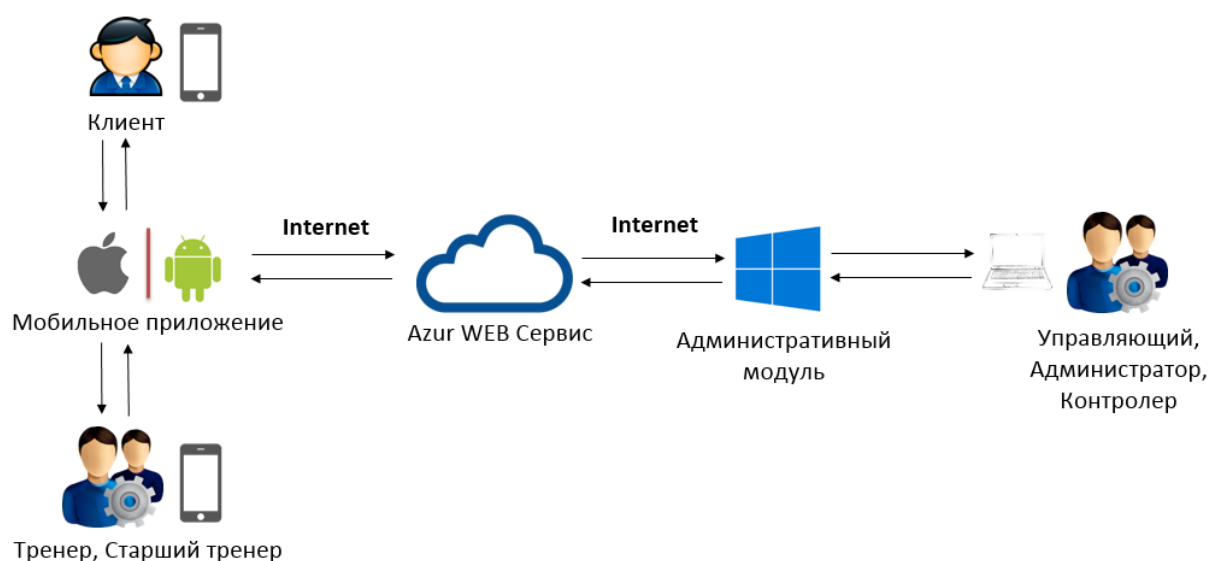


Рисунок 14 – Схема взаємодії компонентів системи

На рисунку 15 зображено взаємодію компонентів системи. При правильному проектуванні архітектури системи повинно бути одне джерело модифікації даних. В випадку розроблюваної системи з базою даних взаємодіє сервіс Web API.

Сервіс взаємодіє з БД за допомогою ADO.NET – технологія підключення та змінення даних в базі даних. Клієнти, адміністративний модуль та мобільні додатки, отримують і змінюють дані використовуючи методи сервісу. Такий підхід інкапсулює структуру БД від клієнтів сервісу та полегшує використання даних розробнику клієнтської частини.

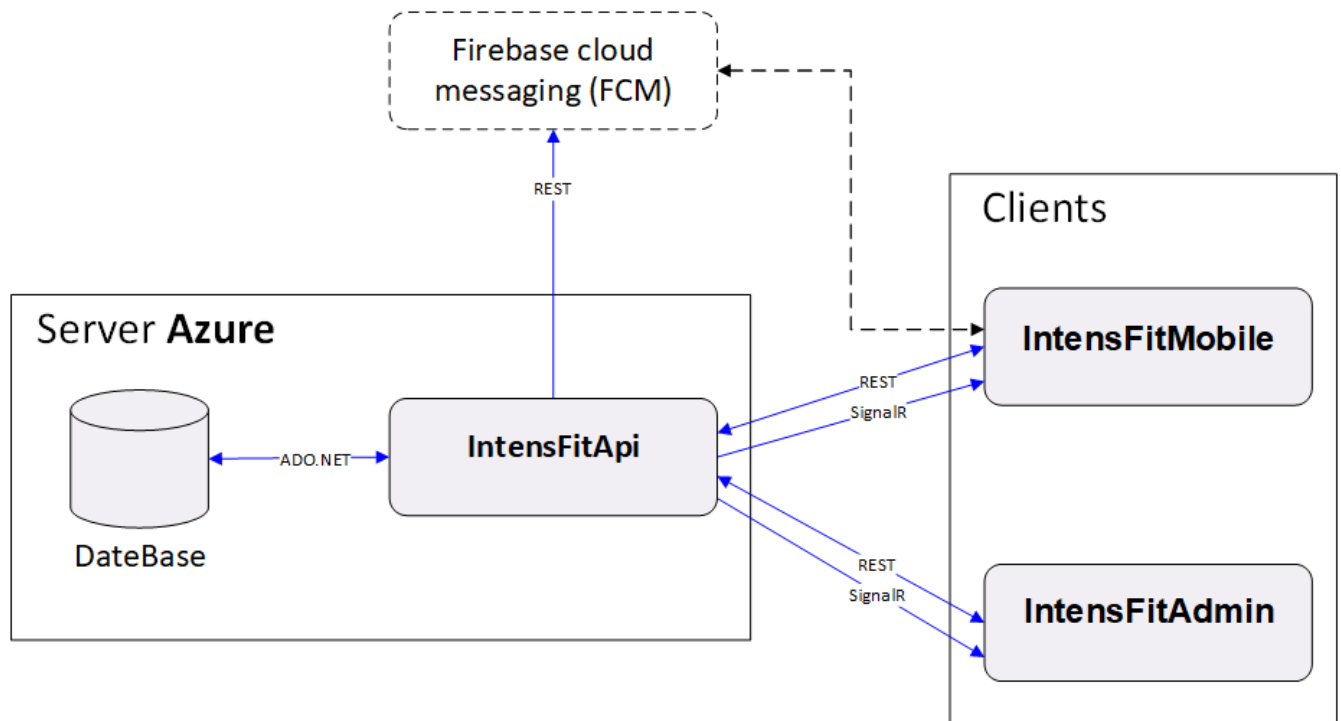


Рисунок 15 – Архітектурна схема взаємодії компонентів

База даних містить велику кількість таблиць, які потрібні для роботи системи. Нижче наведено опис декількох основних сутностей системи. У таблиці 4.1 продемонстрована структура клієнта.

Таблиця 4.1 – Опис таблиці Client

Назва стовпця	Опис	Тип даних	Довжина	Первинний	Зовнішній
id	ID клієнта	int			
createdOn	Коли створений	datetime			
createdById	Ким створений (ID)	int			
modifiedOn	Змінений	datetime			

Закінчення таблиці 4.1

modifiedById	Ким змінений (ID)	int			
timeZoneId	ID часового поясу	int			
countryId	ID країни	int			
city	ID міста	int			
height	Зріст	smallint			
weight	Вага	decimal	6, 1		
artPresSystolic	Тиск систолічний	smallint			
artPresDiastolic	Тиск діастолічний	smallint			
bloodGroup	Група крові	varchar	3		
rhFactor	Резус фактор	char	1		
wakeUpTime	Час пробудження	time	7		
sleepTime	Час сну	time	7		
employmentTypeId	ID зайнятості	tinyint			
employmentFrom	Дата початку зайнятості	time	7		
employmentTill	Дата закінчення зайнятості	time	7		
isRemoved	Видалений	bit			

Тренування – ключова сутність системи, адже основою тренажерних клубів і є тренування. З їх допомогою відвідувач досягає бажаного результату. Поля цієї сутності наведені в таблиці 4.2.

Таблиця 4.2 – Опис таблиці Workout

Назва стовпця	Опис	Тип даних	Довжина	Первинний	Зовнішній
id	ID клієнта	int		X	
createdOn	Коли створений	datetime			
createdById	Ким створений (ID)	int			
modifiedOn	Змінений	datetime			
modifiedById	Ким змінений (ID)	int			
clientSubscriptionId	ID абонементу	int			
statusId	ID статусу	tinyint			
beginDate	Дата початку	datetime			
endDate	Дата закінчення	datetime			
isRemoved	Видалений	bit			
comment	Коментар	nvarchar	100		

4.2 Розробка програмного комплексу

Програмний комплекс складається з двох основних функціональних частин:

- адміністративний модуль – програмний додаток, який виконує функцію керування усією системою в цілому. Налаштування довідників, локалізація, керування анкетами користувачів, ролями доступу, абонементами тощо;

– мобільний додаток – додаток, що забезпечує взаємодію клієнтів та тренерів. Виконання завдань, спілкування, результат досягнення мети, створення та виконання тренувань, додавання фото тощо.

Ці компоненти мають ряд функціоналу, який забезпечує досягнення мети, що поставлена в даній роботі. В той же час два вище описані компоненти являються клієнтами веб-сервісу, що входить до складу системи. Сервіс розміщений в хмарному середовищі Azure (розділ 3.6). Як було описано вище, така технологія допомагає легко та зручно розмішувати свої сервіси та в будь-який момент часу безперебійно виконувати до них запити і отримувати потрібний результат (відповідь).

Сервіс розроблений таким чином, що логічні частини об'єднані в окремі контроллери, які наповнені методами різних типів і при певних вхідних даних повертають очікуваний результат. Структуру розбиття на контроллери відображена в демонстраційному плакаті 3.

Методи контроллера доступні по посиланню через HTTP запит. У клієнта даного сервісу присутнє посилання на Web API сервіс (розділ 3.3), в посиланні вказується назва контроллера до якого звертається клієнт та назва методу в який передаються дані в форматі JSON.

На сервері Azure розміщена база даних розроблюваної системи обслуговування. Клієнти сервісу (адміністративний модуль та мобільні додатки) виконують маніпуляції з даними безпосередньо через даний сервіс. Це дає змогу правильно розподілити архітектуру програмного комплексу для подальшої розробки, масштабування та супроводження.

Веб-сервіс системи має інтеграцію з FCM (розділ 3.9) для можливості відправки Push повідомлень клієнтам. Взаємодія відбувається по створеному токenu (ідентифікатору) для кожного додатку. Токен генерується в момент програмної конфігурації конкретного екземпляру мобільного додатку на конкретному мобільному пристрої. Після цього токен передається через сервіс і

заноситься в базу даних окремим записом. При спробі надіслати повідомлення конкретному користувачу, в базі даних перевіряється наявність такого токєну, після чого безпосередньо відбувається розсилки потрібного Push повідомлень.

Вся, описана вище, загальна архітектура зображена в демонстраційному плакаті 1.

4.3 Профіль клієнта

Форма профілю клієнта представлена нижчу (рисунок 16).

При створенні нового клієнта або відкритті анкети наявного в системі клієнта відображається форма з набором полів та кнопками.

The screenshot shows the 'IntensFit Admin' interface for user 'Пилипчук Саня (380985269967)'. The main content area displays the profile for 'Жигулина А.' (Anna Zhigulina). The form is organized into several sections:

- Client Information:** Includes a photo of a woman, name (Жигулина Анна), date of birth (1981-07-13), age (38), country (Украина), city (Киев), gender (Женской), weight (99,0), height (164), phone (380667079208), email (radioanna@ukr.net), and club (Оболонь).
- Subscription Schedule:** Shows 'Подъем' at 07:00, 'Отбой' at 23:00, and 'Занятость' as 'Свободный гра...' with a 'С' (Saturday) indicator and times 11:00 to 16:00.
- Medical History:** A table of health-related data:

Артериальное давление	120/80	Группа крови	III	Резус фактор	+
Проблемы с давлением	Нет				
Операции	Да		Гинекология год назад.		
Боли в суставах	Да		Протрузии, спондилез, дегенеративные изменения		
Варикозное расширение вен	Да				
Заболевание внутренних органов	Да		Рефлюкс, желч в желудке, жировой гепатоз печени, пролапс митра		
Аллергия на продукты или препараты	Нет				
Гормональные препараты	Нет				
Вредные привычки	Нет				
Занятия голоданием	Нет				
Другие заболевания	Да		Эндометриоз по гинекологии, ежегодный контроль гастро и колоно		
Предпочтения по питанию	Да		Ананас, грейпфрут, кефир		
Дополнительные заметки о здоровье	Нет				
Готовность следовать рекомендациям	Да				
Откуда Вы узнали о нас?	Instagram				
Программа абонента	Интенсив онлайн				

At the bottom right, there are buttons for 'Редактировать' (Edit) and 'Сохранить' (Save). The status bar at the bottom indicates 'Оболонь | Сейчас в клубе: 23 чел' and 'Disconnected'.

Рисунок 16 – Форма профілю клієнта

Для виконання певних дій потрібна наявність відповідних прав у користувача. За допомогою цієї форми можна виконувати такі маніпуляції:

- створювати нового клієнта;
- переглядати анкету;
- редагувати дані клієнта.
- сформувати звіт по клієнтові;
- кнопка переходу в діалог з клієнтом;
- друк анкети клієнта.

Створення профілю клієнта виконується адміністратором і тільки через «Адміністративний модуль» на окремій вкладці «Клієнти» за допомогою кнопки «Створити клієнта».

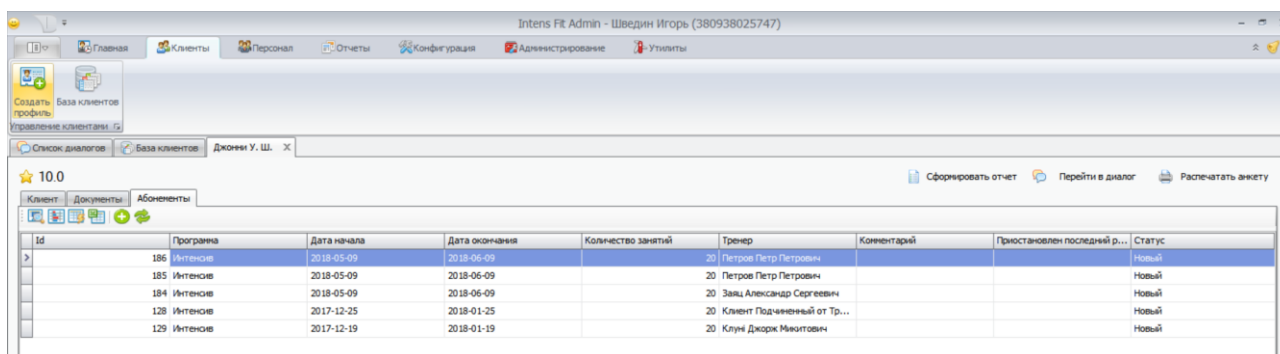
Натиснувши кнопку «Створити клієнта» відкриється вище зображена екранна форма, яка містить набір основних параметрів, які потрібно заповнити. Форма складається з наступних блоків:

- основні дані;
- фото профілю;
- зайнятість клієнта та розклад сну;
- тиск, група крові та резус фактор;
- запитання/відповіді щодо стану здоров'я клієнта;
- платіжний календар (періоди оплати, статус);
- прикріплення файлів (довідки від лікаря, анкета, інші документи).
- завантаження та редагування цих документів;
- додавання абонементу;
- програма абонементу;
- постановка мети;
- запитання про джерело, звідки клієнт дізнався про клуб (для статистики та планування маркетингу).

При відкритті анкети створеного клієнта всі поля заблоковані від редагування, тобто ніяких змін в анкеті виконати неможливо. Змінити дані можна

тільки після натиснення кнопки «Редагувати». Для збереження змін обов'язкове натиснення кнопки «Зберегти» (збереження змін в базу даних).

Форма профілю клієнта включає дочірні вкладки, де розташовані вкладки «Абонементи» і «Документи» клієнта (рисунок 17). На екранній формі абонементів зображений весь список абонементів, в тому числі не активні, завершені тощо.



Id	Програма	Дата початку	Дата закінчення	Кількість занять	Тренер	Коментарій	Встановлено останній р...	Статус
186	Итениев	2018-03-09	2018-06-09	20	Петров Петр Петрович			Новый
185	Итениев	2018-05-09	2018-06-09		Петров Петр Петрович			Новый
184	Итениев	2018-05-09	2018-06-09		Защ Александр Сергеевич			Новый
128	Итениев	2017-12-25	2018-01-25		Клиент Подключенный от Тр...			Новый
129	Итениев	2017-12-19	2018-01-19		Клинт Джорж Михайлович			Новый

Рисунок 17 – Представлення списку абонементів Клієнта

В абонементі виконуються налаштування та вказуються наступні параметри:

- вибір тренера;
- встановлення періоду занять (дати та кількість занять);
- в автоматичному режимі створення діалогу та додаванням до нього учасників;
- вибір часу для різних прийомів їжі;
- вибір часу відправлення ваги;
- вибір часу відправлення нагадувань про виконання певних дій.

На формі створення абонементу присутня кнопка активації абонементу в потрібний час. Також, як і на майже всіх формах доступне редагування параметрів та збереження їх в базу даних.

Категорія абонементу клієнта (онлайн чи офлайн) буде визначатися автоматично в залежності від обраної програми тренувань при створенні абонементу. Наприклад, «Інтенсив онлайн» або «Інтенсив офлайн».

На формі «Документи» додаються файли клієнта (пункт 4.6). Файли – це медичні справки, рекомендації по харчуванню тощо. Є можливість завантажити на свій комп'ютер прикріплені файли по одному або ж всі одразу.

Форми, які мають табличне представлення даних, містять функціонал, за допомогою якого є можливість групувати, сортувати, фільтрувати та здійснювати пошук за певними умовами, виконувати експорт даних таблиці в файл на комп'ютер.

Вигляд профілю клієнта зображений на рисунку 18.

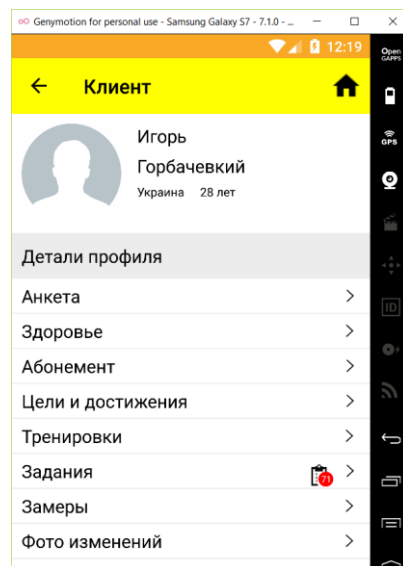


Рисунок 18 – Форма профілю Клієнта в мобільному додатку

До форми профілю поставлені такі вимоги:

- створення та редагування основних блоків;
- додавати та змінювати абонемент;
- прикріпляти та завантажувати документи тощо.

Дії, які може виконувати клієнт в системі, зображені на так званій Use case діаграмі (демонстраційний плакат 4).

4.4 Профіль тренера

На окремій вкладці адміністративного модуля «Тренери» та натисненні кнопки «Створити тренера» відкривається форма, яка зображена на рисунку 19.

The screenshot shows the 'IntensFit Admin' interface. The main content area displays a form for creating a trainer profile. The form is titled 'Список тренеров' and shows a profile for 'Йохансен С.'. The form fields are as follows:

Тип тренера	Тренер
Фамилия	Йохансен
Имя	Скарлет
Отчество	
Пол	Мужской
Дата рождения	1995-08-10
Возраст	24
Телефон	380501045865
Email	
Скоре	
Клуб	Печерок

Additional elements include a photo upload section with a 'Обрезать фото' button, and action buttons 'Редактировать' and 'Сохранить' at the bottom right. The status bar at the bottom indicates 'Печерок | Сейчас в клубе: 23 чел' and 'Disconnected'.

Рисунок 19 – Профіль тренера

Форма профілю тренера подібна структурою до вище описаної форми клієнта, але з меншою кількістю полів та складається з таких основних параметрів:

- ім'я;
- фото;
- посада;

- номер мобільного телефону, адрес e-mail скриньки тощо.

Аналогічно формі клієнта, наявна можливість (за наявністю відповідних прав) виконувати:

- створення тренера;
- перегляд;
- редагування анкети;
- друк анкети;
- формування звіту;
- кнопка переходу в діалог.

Для збереження анкети потрібно обов'язково заповнити всі необхідні поля.

Необхідні функціональні можливості профілю:

- створення та редагування основних параметрів анкети тренера;
- створення профілю тренера доступно тільки через Адміністративний

модуль.

Анкета даного типу користувача в системі пов'язана з сутністю абонементу, який, в свою чергу, відноситься до конкретного клієнта. Як наслідок, профіль тренера відноситься до конкретних діалогів, тренувань, оцінок, замірів тощо. Після створення менеджером анкети тренера, користувач має можливість авторизуватися мобільного додатку за допомогою номеру телефону.

Функціональні можливості користувача типу тренер описані на Use case діаграмі (демонстраційний плакат 4).

4.5 Ролі в системі

Для кожного створеного користувача системи повинні вказуватися ролі за допомогою відповідного функціоналу (дивитись нижче). Управління ролями виконується в Адміністративному модулі на вкладці «Адміністрування».

Зображена форма (рисунок 20) розділена на дві частини табличного представлення. Зліва знаходиться список користувачів, а справа список ролей в системі. Для призначення або відміни ролі потрібно відмітити її. Збереження в базу даних відбувається при натисненні галочки і про це сигналізує повідомлення, яке з'являється в правому нижньому куті.

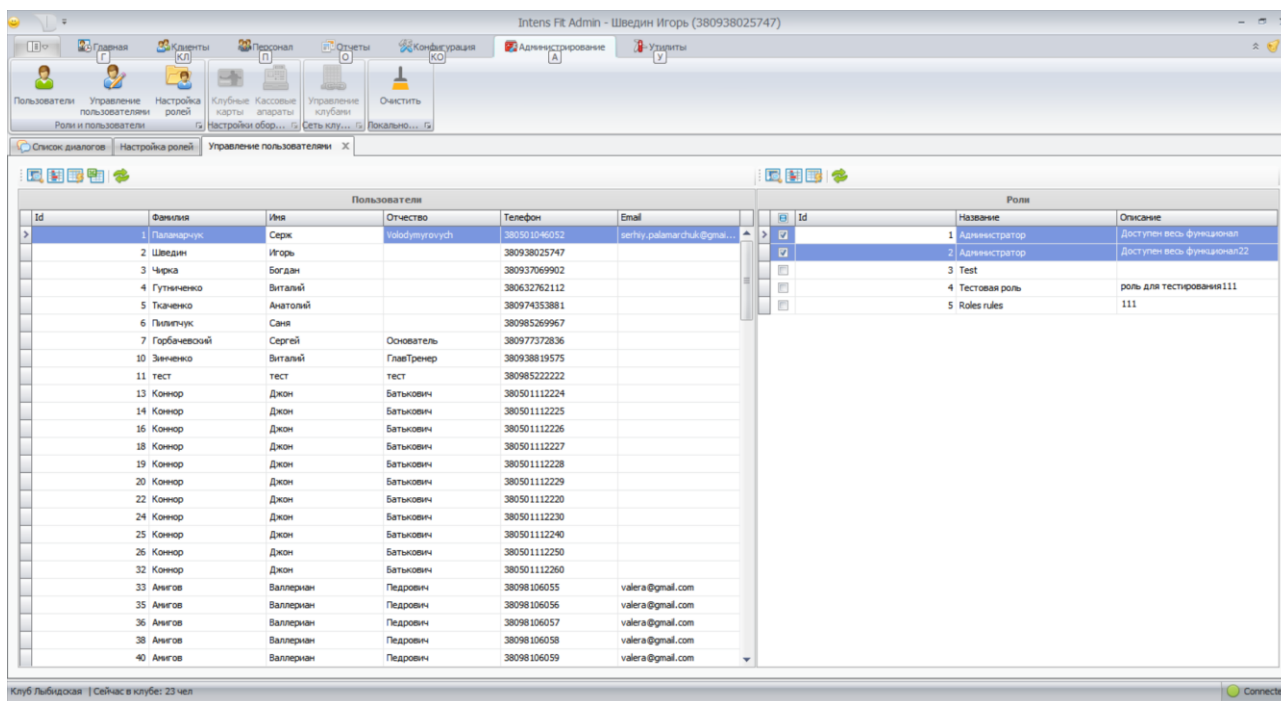


Рисунок 20 – Призначення ролей користувачам

Призначена роль визначає доступність того чи іншого функціоналу певної частини системи (таблиця 4.3).

Таблиця 4.3 – Розподілення ролей

	Адміністративний модуль	Мобільний додаток	Коментарі
Адміністратор	+	-	
Керуючий	+	+	Старший тренер в мобільному додатку
Контролер	+	-	
Тренер	-	+	
Клієнт	-	+	

Кожна роль включає в собі набір функціональної частини, який доступний для неї. На рисунку 21 зображена форма, в якій міститься таблиця з ролями (ліворуч) та таблиця з доступним функціоналом для цієї ролі (праворуч). Доступний функціонал – це програмні об’єкти, які включають в собі назву, групу і зрозумілий опис. Для додавання певної функціональності для певної ролі потрібно відмітити галочкою той чи інший елемент в правій частині.

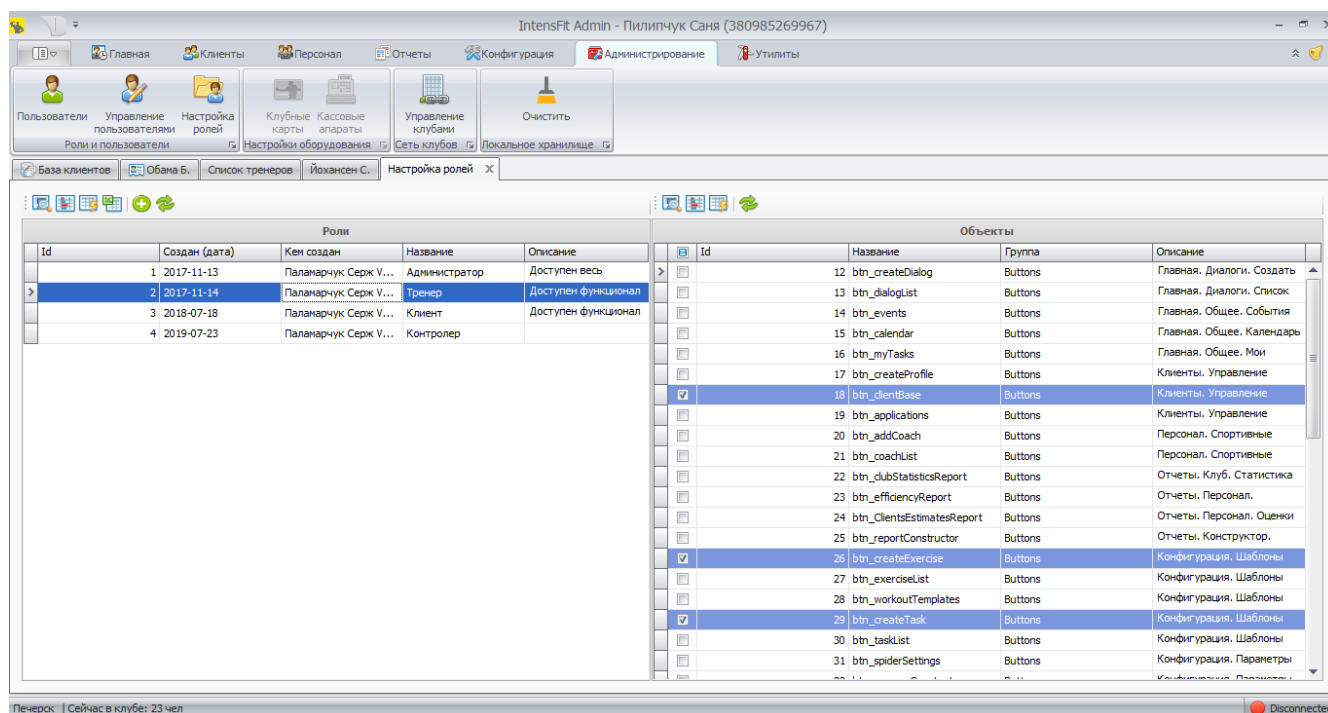


Рисунок 21 – Форма налаштування певного функціоналу для ролей

4.6 Прикріплення файлів

В системі наявний механізм, що забезпечує можливість прикріплення файлів для кожного клієнта системи. Файли можуть бути прикріплені у вигляді медичних довідок, рекомендацій по харчуванню тощо. Це дає можливість більш ефективно та правильно підлаштовувати програму тренувань та харчування кожному клієнту, що в результаті допоможе швидше отримати результат.

Реалізована можливість видалення та експорту прикріплених файлів (рисунок 22).

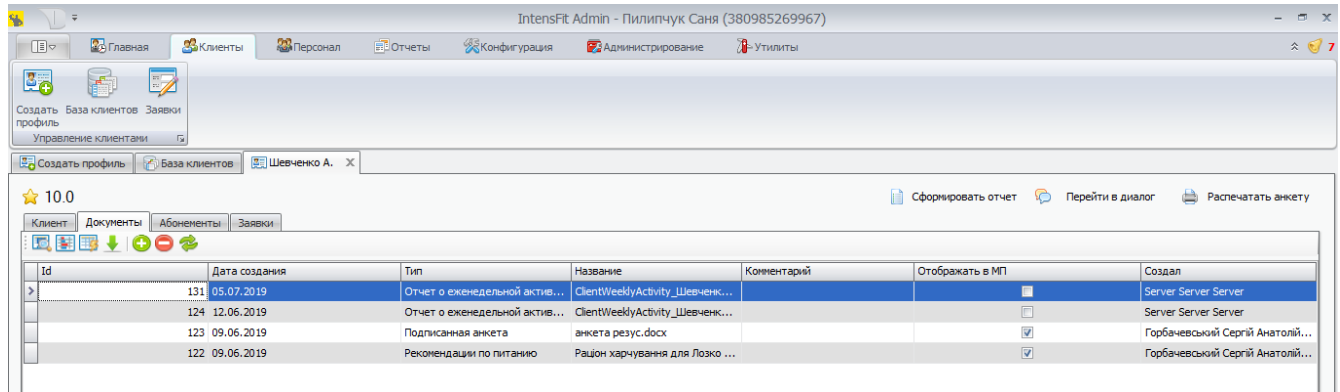


Рисунок 22 – Форма файлів клієнта

4.7 Абонемент

Налаштування абонементу виконується в Адміністративному модулі (рисунок 23). В автоматичному режимі активація абонементу відбувається на один місяць. Для цього періоду клієнтові надаються двадцять занять (згідно технічного завдання до продукту):

- якщо по закінченню місяця клієнт виконав менше вказаних 16 тренувань, то абонемент продовжується на потрібний період;
- автоматичне закінчення абонементу проходить в той момент, якщо по закінченню місяця у клієнта виконано більше 16 тренувань.
- наявна можливість активувати абонемент (по суті розпочати абонемент).

На зображеній формі вище присутня можливість редагувати параметри абонементу, такі як: зміна кількості занять або період дії. Присутня можливість змінювати тренера для абонементу (наприклад, у зв'язку з хворобою тренера тощо).

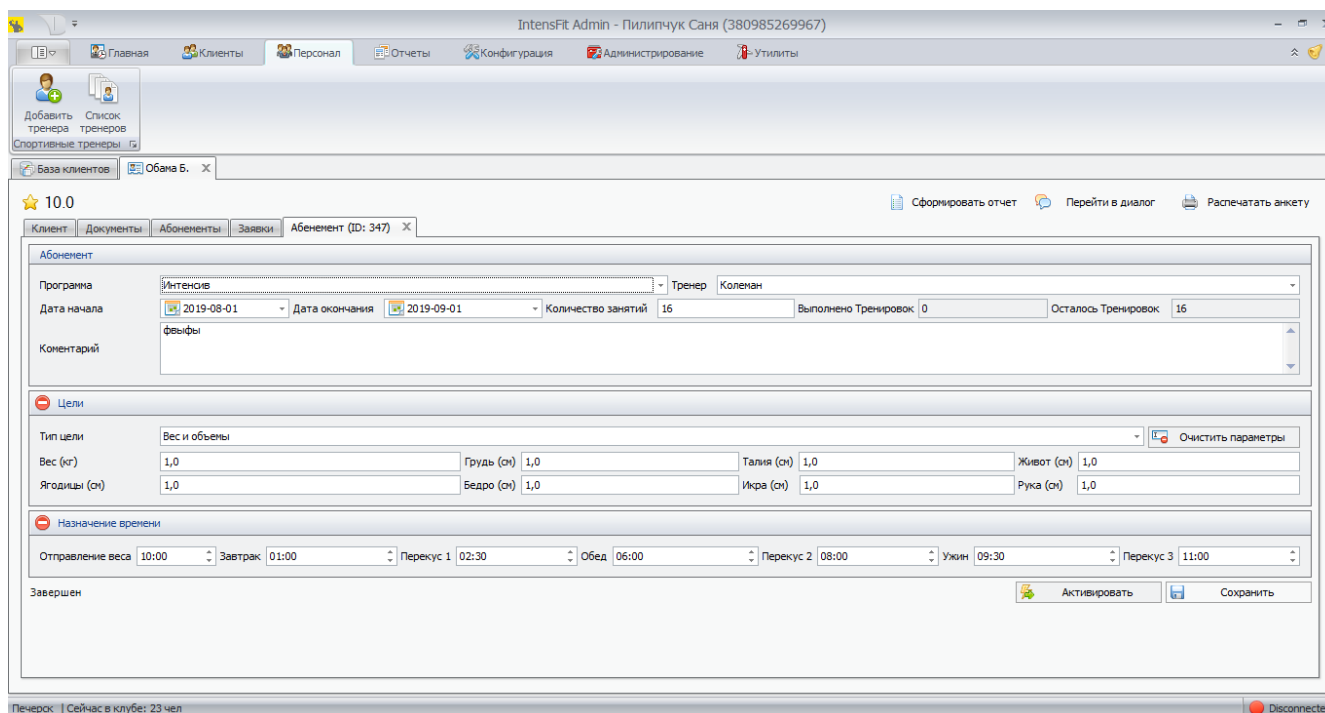


Рисунок 23 – Створення абонементу

По приходу строку закінчення дії абонементу, а саме за 5 днів, для тренера створюється задача, яка звертає його увагу на даного клієнта та, можливо, необхідність виконати певні дії. Клієнтові додається задача за 3 дні до закінчення абонементу, яка так же само сповіщає його про можливе виконання певних дій.

Після створення абонементу та переведення його в активний стан активується повний набір функціональності системи для активної взаємодії користувачів та виконання поставлених завдань. Активний абонементу може бути тільки один в розрізі клієнта.

Головний та важливий параметр, який потрібно вказати при створенні абонементу – це мета, ціль клієнта про досягнення певних результатів впродовж використання абонементу шляхом виконання тренувань, завдань та різних вказівок збоку тренера. Також на цій формі налаштовується розклад відправки ваги або прийому їжі клієнтом. За вказаним часом будуть надсилатися на мобільний додаток Push повідомлення для нагадування.

4.8 Діалог

Для користувачів адміністративного модуля з відповідними ролями наявна функціональна можливість відображення всіх створених діалогів в системі. Це можуть бути автоматичні діалоги клієнта з тренером або групові діалоги (містять більше, ніж два учасника).

При цьому наявні наступні можливості:

- створення нових діалогів;
- перегляд діалогів;
- редагування назви та фото діалогу;
- додавання та видалення учасників діалогу;

Діалог (рисунок 24) – це з одна основних сутностей системи. В рамках цієї сутності реалізований механізм комунікації клієнта з тренерським складом. Діалог розроблений у вигляді чату між двома або більше учасниками. Основні учасники діалогу – це тренер і клієнт.

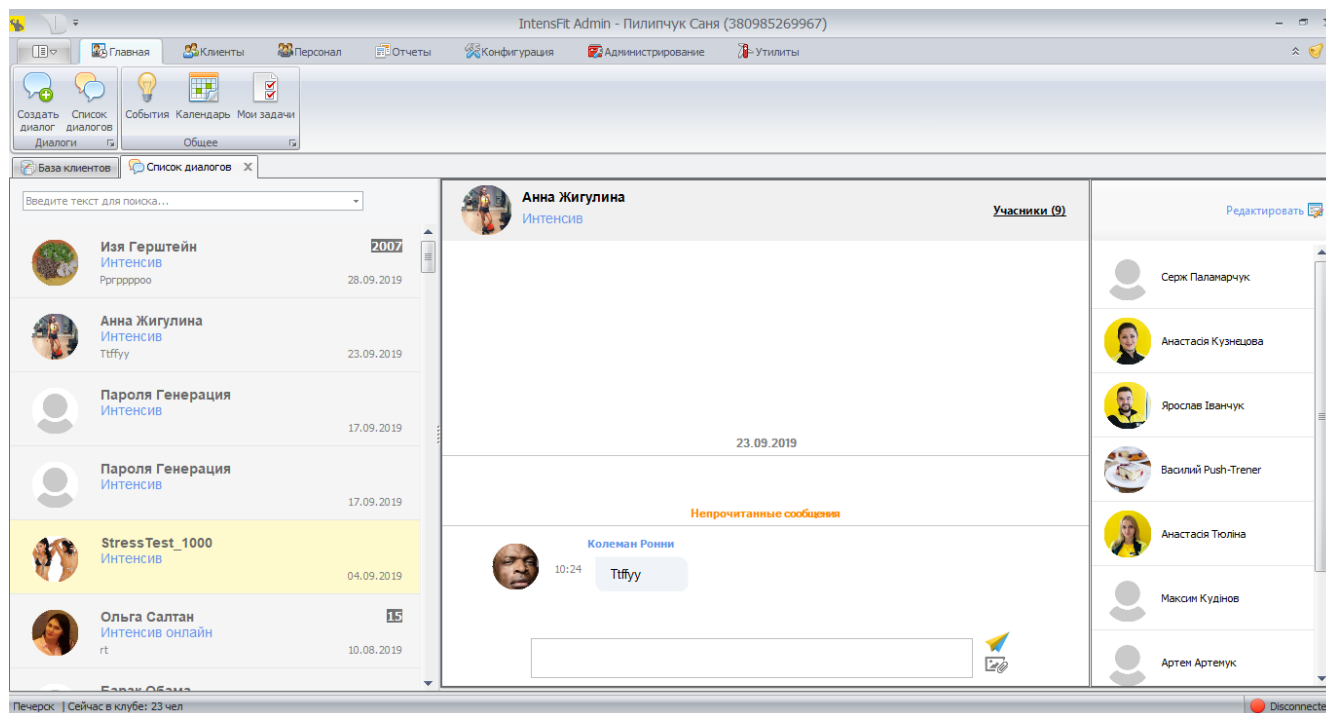


Рисунок 24 – Форма списку діалогів та діалогу

В момент створення абонементу відбувається створення відповідного діалогу з додаванням клієнта, для якого створюється абонемент, та тренера, який було призначено для цього абонементу. Або ж доступне ручне створення діалогу та додавання будь-якої кількості учасників. Створений діалог для кожного користувача, який доданий до нього, з'явиться мобільному додатку (рисунок 25).

Діалог має такі функціональні особливості:

- друкування і відправлення текстових повідомлень, повідомлень із зображеннями між учасниками;

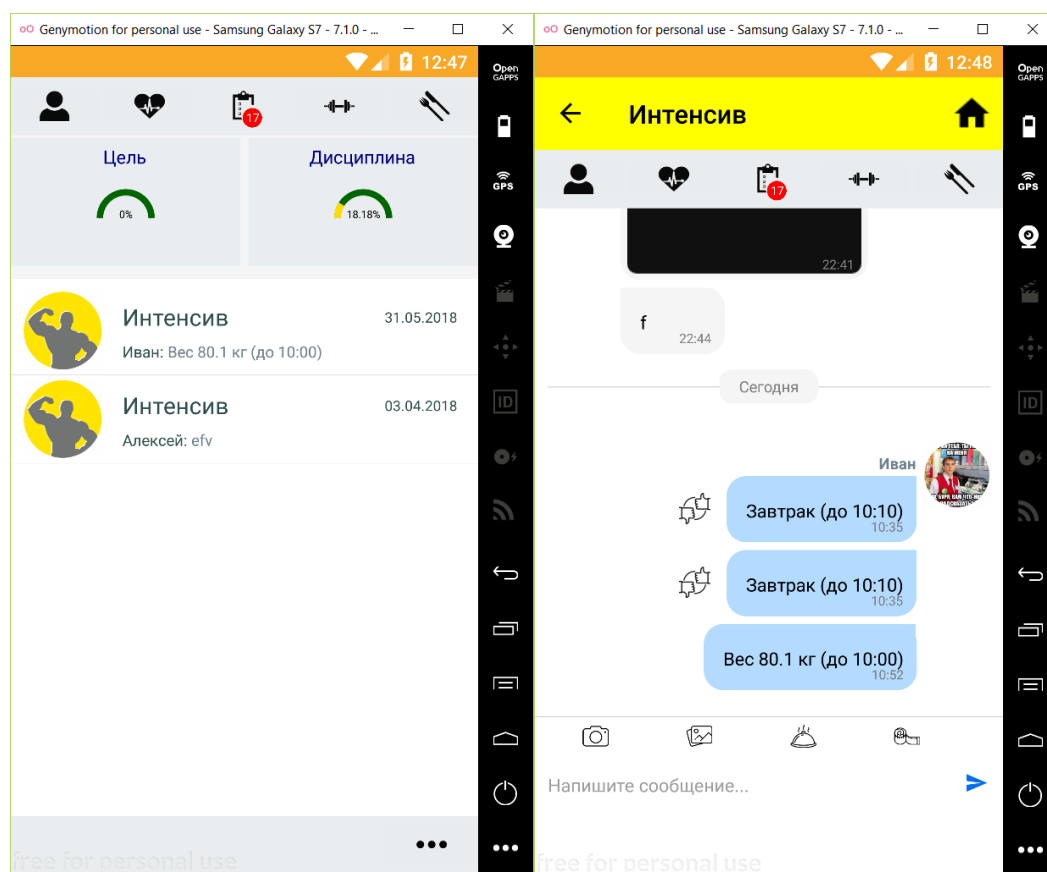


Рисунок 25 – Список діалогів та область діалогу в мобільному додатку

- містить різні типи повідомлень, включаючи ті, що створені в автоматичному режимі чи користувачем в результаті виконання завдань. Для прикладу клієнт, виконуючи завдання відправки ваги або конкретного прийому

їжі може прикріпляти фотографію та додавати опис. Такий тип повідомлення містить можливість оцінки тренером (зображено на рисунку 25). Вся інформація зберігається в базі даних та в подальшому буде використовуватися в побудові звітів та графіків для візуальної оцінки результатів роботи, як клієнта, так и тренера;

- автоматична відправка звітів (PDF файл) в діалог за певний визначений період;

- історію діалогу можна побачити прокрутивши сторінку діалогу вверх.

Користувачеві з роллю тренера відображаються тільки діалоги з його клієнтами; старший тренер може бачити діалоги тренерів, якими він керує; користувач, який має роль керуючого бачить всі діалоги системи.

4.9 Broadcast діалог

Broadcast діалог (груповий діалог) – це окремий тип діалогу, який створюється за принципом «один» до «багатьох». Цей тип діалогу може включати будь-яку кількість учасників які зареєстровані в системі. Створення такого типу діалогу можна побачити відповідно до рисунка 26 – додавання назви діалогу, зображення та вибір учасників для включення в створюваний діалог.

Після створення діалогу є можливість переглядати учасників діалогу на панелі праворуч, переходити в анкету кожного учасника натисканням імені або фото профілю, кнопка редагування учасників діалогу в будь-який час.

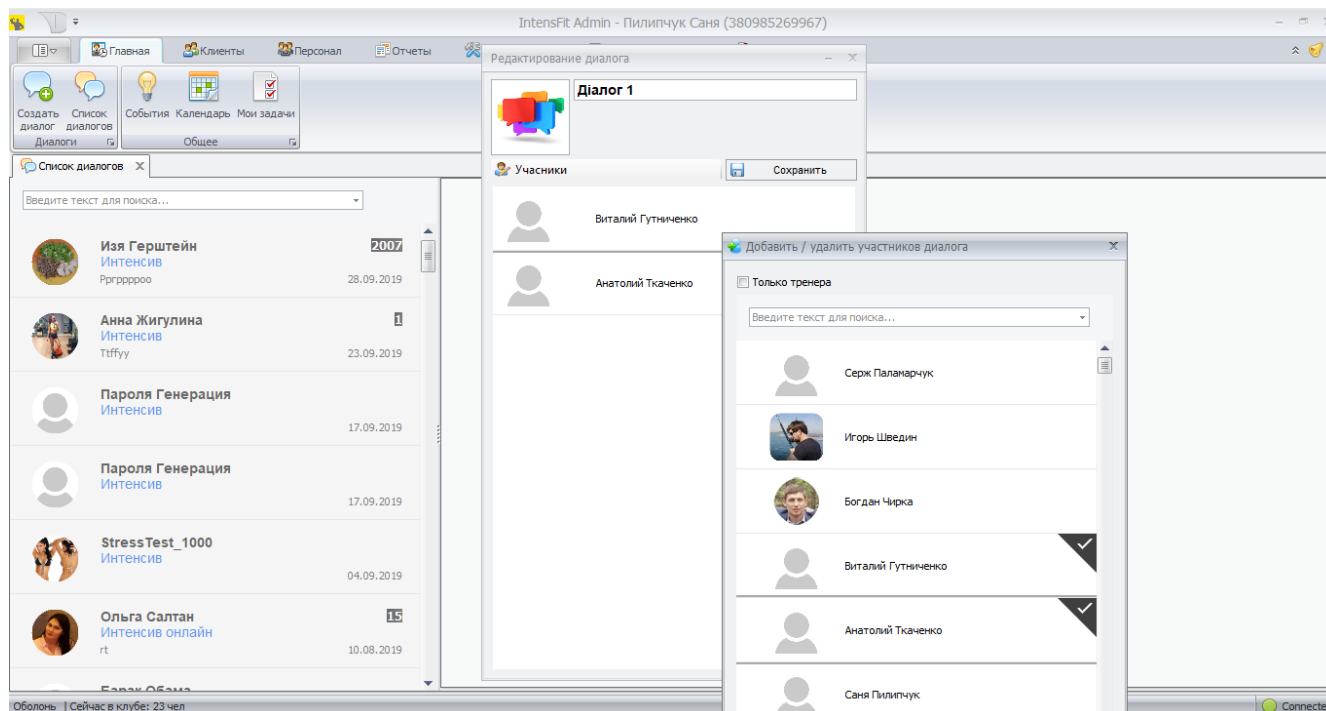


Рисунок 26 – Форма створення діалогу та додавання учасників

4.10 Мета

Мета – це те, що хоче досягти клієнт. В розробленому рішенні – це фізичні параметри, такі як: збільшення чи зменшення ваги тіла або ж зміна обсягів конкретних частин тіла (руки, ноги, талія, сідниці тощо). Мета визначається після співбесіди адміністратором з клієнтом та встановлюється на певний період (так званий період абонементу) в адміністративному модулі. Адміністратор має можливість видаляти чи змінювати поставлену мету клієнта.

Наявні такі типи мети:

- зміна тільки ваги тіла;
- зміна тільки обсягів (параметрів) тіла;
- зміна як ваги, так і обсягів тіла;
- підтримки наявної форми клієнта.

Параметри для задання мети описані в таблиці 4.4.

Таблиця 4.4 – Параметри мети

Параметр	Од. вимір.
Рука	см
Ікра	
Живіт	
Стегно	
Груди	
Сідниці	
Талія	
Вага	кг

Вказавши мету, її можна побачити в мобільному додатку перейшовши на відповідну форму з профілю клієнта, де можна також відслідковувати зміни та досягнення цієї мети. В будь-який час клієнт або тренер має можливість переглядати екрану форму в мобільному додатку, де відображається задана мета для клієнта і результати, які були внесені в програму протягом тренувань в рамках абонементу. За допомогою цього функціоналу можна одразу бачити результат клієнта і, якщо потрібно, коригувати кількість тих чи інших тренувань, завдань для досягнення поставленої мети.

4.11 Заміри клієнта та досягнення мети

Заміри – це ряд параметрів, які створюються і зберігаються в системі одночасно. Перша частина замірів заноситься в момент першого відвідування тренажерного клубу клієнтом. За допомогою цих параметрів відбувається аналіз прогресу змін. Перед початком першого зняття з тренером відбувається запис замірів в мобільному додатку. Далі по мірі занять клієнта тренер з певною періодичністю, наприклад, один раз в тиждень, робить заміри клієнта за основними параметрами, які були вказані при встановленні мети (див. розділ 4.10) (вага, ноги, сідниці, талія, руки) і їх збереження. При регулярному внесенні цих показників виконується наступне:

- визначається індикатор досягнення поставленої мети клієнтом у відсотковому відношенні;
- також в мобільному додатку будуються графіки досягнення конкретної мети в залежності від поставлених цілей. При перегляді графіків наявна можливість фільтрування по кожній меті або за обраний часовий період;
- після того, як було занесено заміри за певний період, буде створюватися автоматичний звіт з результатами досягнення в форматі PDF і відправлятися клієнту в діалог;
- клієнти у яких буде низький результатами виділяються спеціальним «маркером», щоб підвищити увагу тренера або керуючого до цього клієнта задля виконання певних дій і для підвищення ймовірності досягнення поставленої мети.

У мобільному додатку можна переглянути всі заміри клієнта в рамках одного абонементу та їх зміни в результаті виконання тренувань, щоб визначити процент досягнення поставленої мети. На екрані, зображеному на рисунку 27, зображені початкові заміри клієнта, заміри, які були поставлені для досягнення, і результат змін. Результат підсвічується зеленим кольором при позитивних змінах або червоним – при негативних. На екрані справа, зображеного на тому ж рисунку 27, проілюстрована візуалізація зміни конкретного параметру. Під графіком розміщено журнал записів конкретних точкових змін, які додаються тренером в відповідні проміжки часу.

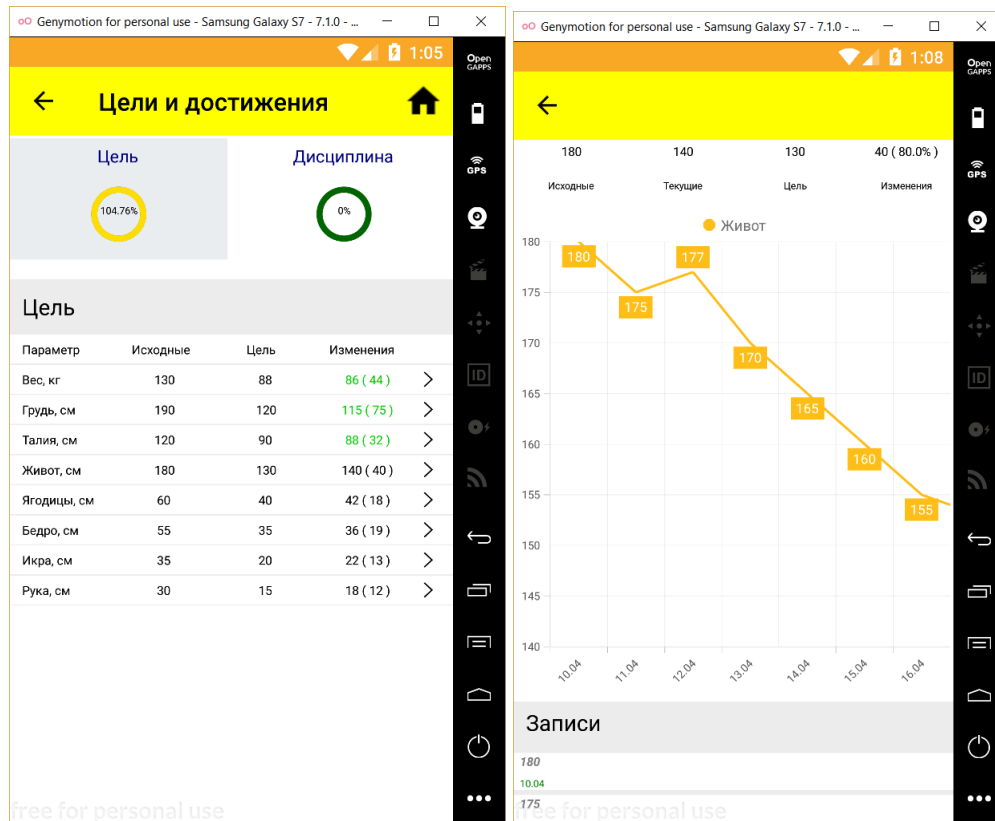


Рисунок 27 – Цілі та їх візуалізація на графіку

Розрахунок досягнення мети починається від першого внесення замірів клієнта на першому тренуванні до останніх замірів при закінченні тренувального циклу. Заміри вносить тренер власноруч в мобільному додатку. Розраховуючи індикатор досягнення мети, потрібно врахувати:

- якщо задана мета тільки вага, тоді 100% орієнтація на вагу;
- якщо задані тільки обсяги тіла, тоді 100% орієнтація на обсяги;
- якщо об'єми тіла та вага, тоді вплив на результат ваги 70%, обсягів 30%.

Для кожного клієнта розраховується індикатор досягнення поставленої мети в відсотковому співвідношенні. Клієнта бачить даний індикатор досягнення мети у вигляді графіка та результуючими відсотками. Для тренера такий індикатор розміщений на верхній панелі екрану в діалозі з відповідним клієнтом.

4.12 Дисципліна

Дисципліна клієнта визначається в залежності від виконання всіх необхідних завдань таких як:

- відправка прийому їжі (кожен день);
- прийом рекомендованої їжі (тренер оцінює відправлений прийом їжі в діалозі);
- зважування та відправка ваги (кожен день);
- відвідування тренувань (згідно розкладу), наприклад, за програмою Інтенсив, це не менше чотирьох тренувань на тиждень;

Клієнт або тренер має можливість переглянути виконання тих чи інших завдань у мобільному додатку в числовому представленні і у вигляді графіку.

Система виконує аналіз активностей клієнта за кожен день: наявність відправки клієнтом різних прийомів їжі, ваги, виконання завдань та оцінка параметрів тренером.

4.13 Тренування

Тренування – це сутність в системі, в яку входить наступний набір характеристик:

- 1) перелік стандартних вправ;
 - а) кожна вправа має підходи та повторення;
 - б) можлива наявність окремого блока, який називається «суперсет». Він включає в себе набір вправ, які мають підходи та повторення, з черговою виконання вправ;
 - в) деякі вправи включають тільки час їх виконання;
- 2) є можливість створювати власні вправи. Такого типу вправи тренер створює власноруч, коротко описавши вправу, прикріплює фото правильного

виконання вправи, та/або прикріплює посилання на відео з правильним виконанням вправи;

3) вправа може мати статус – нова, активна і виконана.

На рисунку 28, зліва, зображено список тренувань клієнта (створених тренером). Тренування згруповані в рамках кожного абонементу, що дозволяє згортати кожну групу. З правої частини рисунка 28 зображено форму вправи, де знаходиться саме тренування, його назва, можливість переглянути відео для правильного виконання вправи, кількість підходів, кількість повторів та вагу. До тренування тренер може додавати коментар та вказувати відсоткове виконання вправи. Для іншої групи вправ можливо вказувати час виконання.

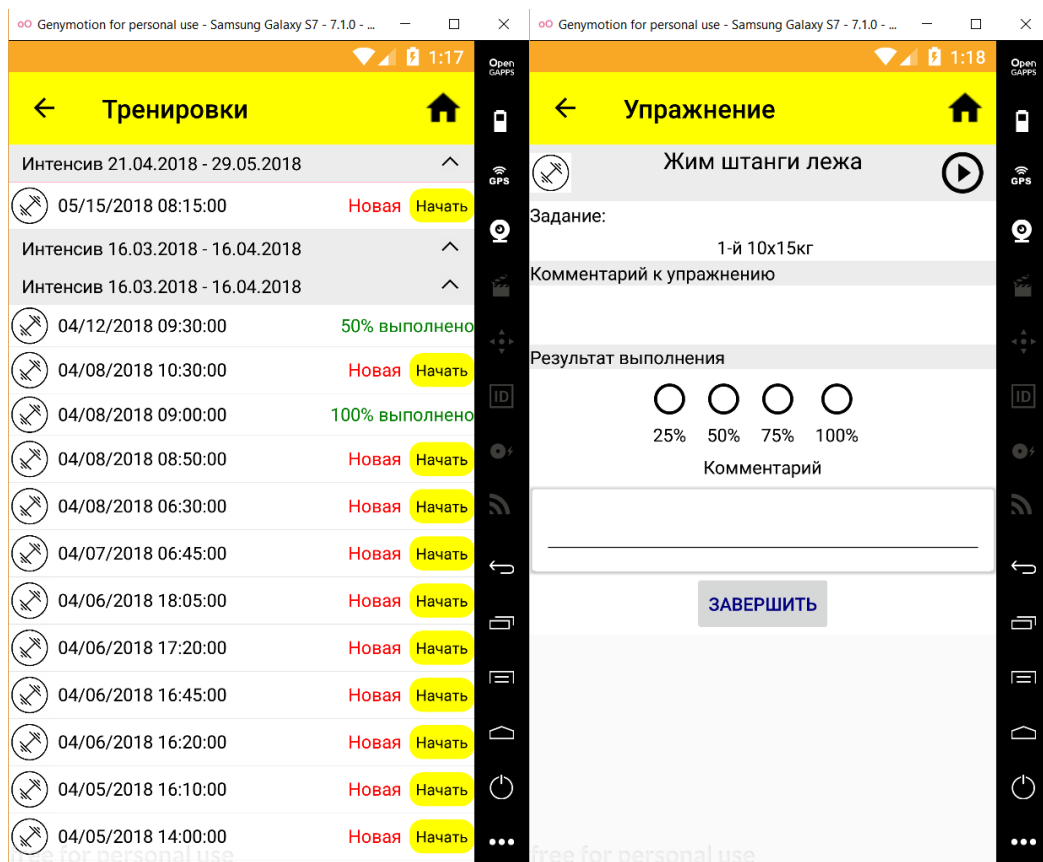


Рисунок 28 – Список тренувань та тренування

Створення тренування (рисунок 29):

- тренування створює тренер, обирає перелік вправ та додає кількість підходів і повторень;
- створене тренування додається для конкретного абонемента;
- у тренера є можливість змінювати вправи в процесі виконання клієнтом тренування.

На формі (рисунок 29) зображено створення нового тренування клієнту, вказує дату тренування та час виконання, коли клієнт повинен прийти в тренажерний зал і виконати його. Можливо вказувати коментар для тренування, щоб вказати деталі.

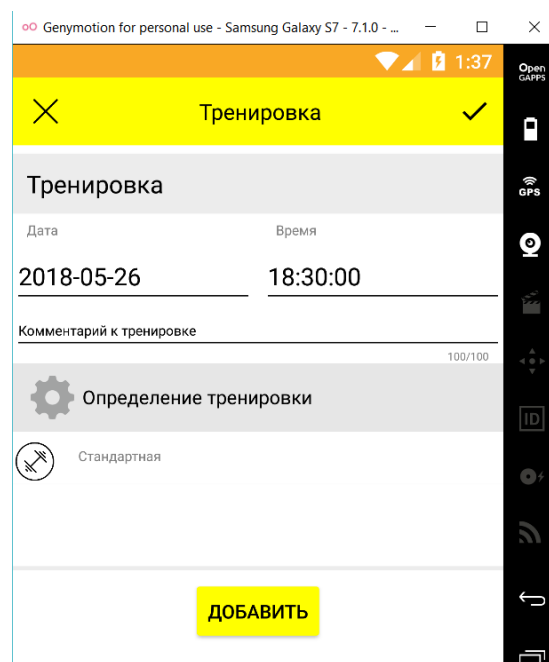


Рисунок 29 – Задання параметрів тренування

На наступному кроці вибирається тип тренування – «Стандартне» чи «Кругове». Перший тип – це тренування, де вказується послідовність вправ для виконання з кількістю підходів, повторень тощо. Другий – це тренування, яке

містить послідовність вправ, кожне з яких містить підхід з визначеним часом, вагою тощо.

На рисунку 30 зображений процес наповнення тренування вправами.

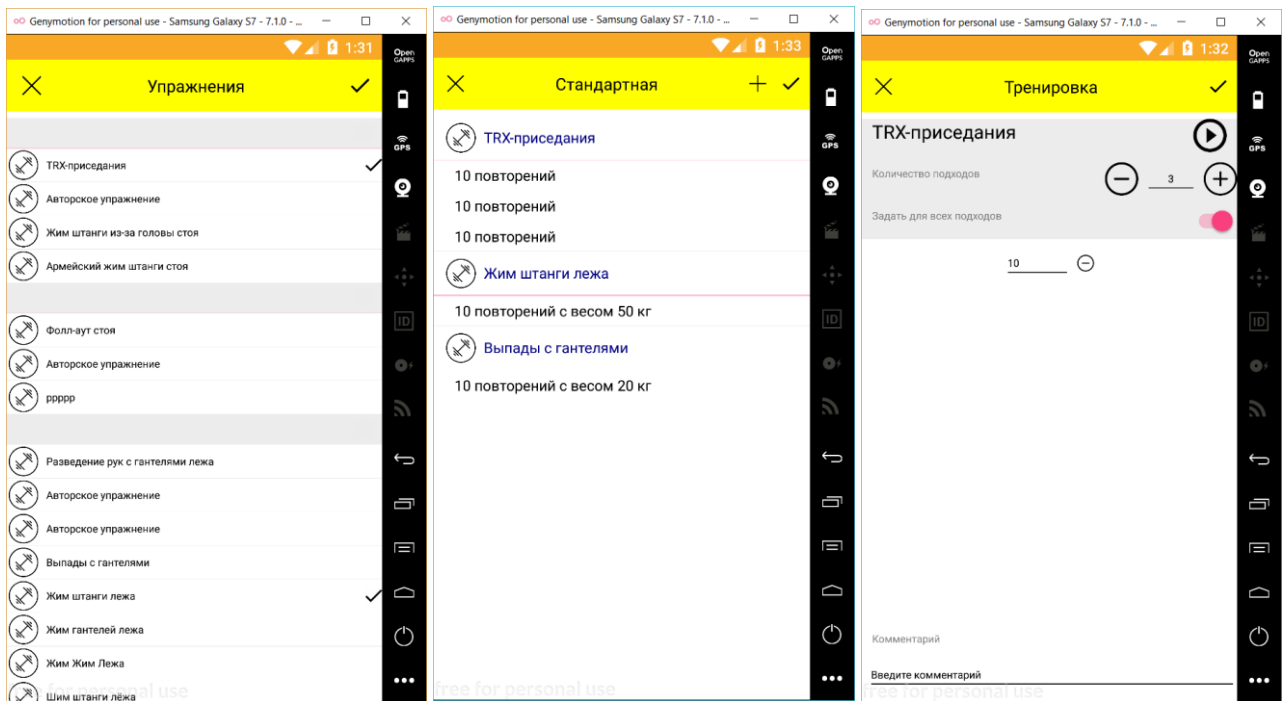


Рисунок 30 – Створення тренування

Відмічаються потрібні вправи та вказуються потрібні параметри (повтори, кількість підходів, час тощо) для кожного.

Додавання вправ (наповнення довідника) для вибору відбувається за допомогою функціоналу в адміністративному модулі (рисунок 31).

Процес виконання тренування:

- виконуючи тренування клієнт відмічає кожну вправу, яку виконав клієнт та відзначає результат цього виконання;
- якщо вправа не була виконана або виконана не в повному обсязі, то клієнт має можливість проставити нульовий відсоток виконання вправи.

Завершення тренування:

- після виконання та задання результатів кожної вправи, клієнт повинен завершити тренування дією «Виконана»;
- клієнт має можливість вказати самопочуття під час або після тренування;

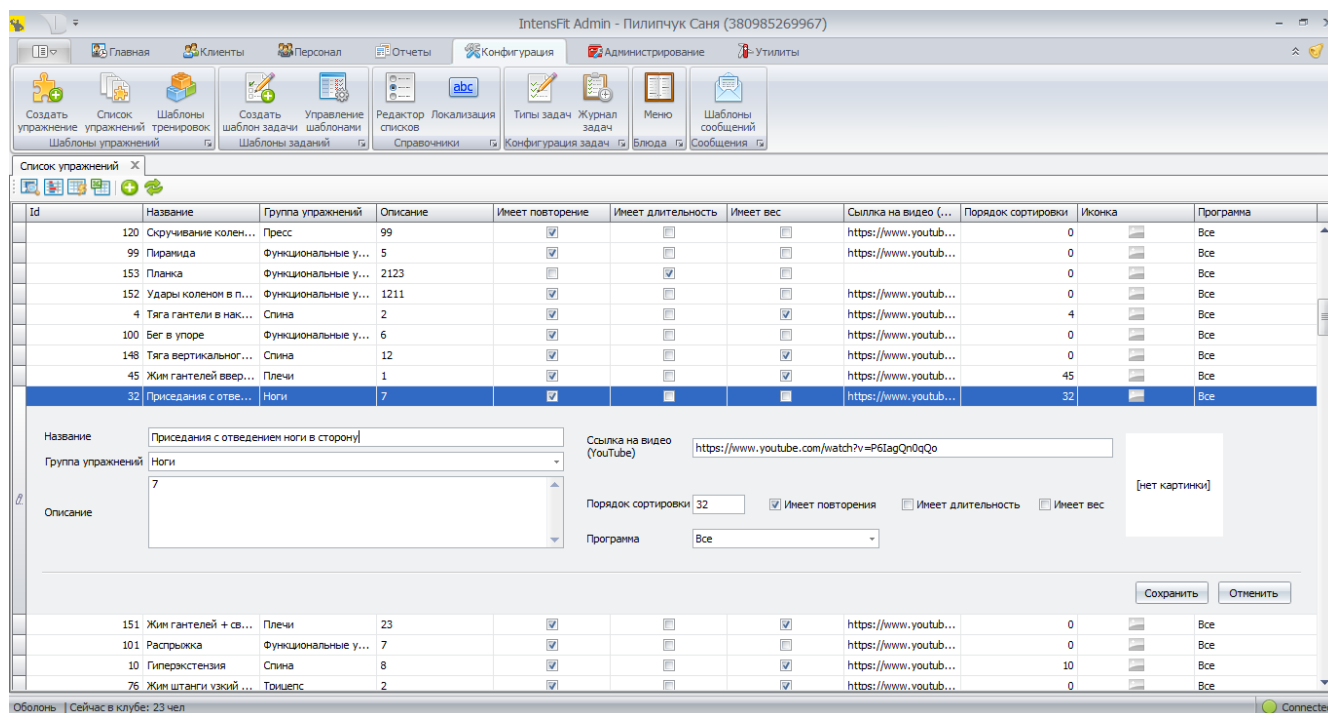


Рисунок 31 – Форма додавання та редагування вправ

- тренер повинен підтвердити, що тренування насправді було виконане;
- якщо клієнт пропустив призначене тренування, то тренер має можливість вказати статус «Не виконано» із зазначенням причини або перенести тренування на наступну дату/час;
- тренування, які були не виконані, помічені «маркером» для фокусу уваги в звітах. Тренеру не нараховується заробітна плата, якщо тренування відмічено зі статусом «Не виконано».

Для кожного клієнта в системі зберігається історія тренувань клієнта. Клієнт має можливість переглядати історію власних тренувань. Історія тренувань можливо приховати для неактивних клієнтів.

4.14 Шаблони тренувань

Тренери повинні працювати одночасно з багатьма клієнтами. Для кожного клієнта потрібно створити тренування, наповнити кожне тренування вправами з підходами, кількістю повторів тощо (рисунок 32). Задля прискорення і полегшення роботи тренерів передбачено функціонал «Шаблони тренувань». Такі шаблони включають в себе набір передбачених тренувань з вправами та кількістю повторів, ваги тощо в залежності від типу вправи. По суті це готові тренування, які залишається тільки призначити клієнтові для конкретного абонементу.

The screenshot shows the 'Шаблоны тренировок' window in the IntensFit Admin application. The window title is 'Шаблоны тренировок'. The interface is divided into several sections:

- Menu Bar:** Includes 'Главная', 'Клиенты', 'Персонал', 'Отчеты', 'Конфигурация', 'Администрирование', and 'Утилиты'.
- Toolbar:** Contains icons for 'Создать упражнение', 'Список упражнений', 'Шаблоны тренировок', 'Создать шаблон задачи', 'Управление шаблонами', 'Редактор списков', 'Локализация справочников', 'Типы задач', 'Журнал задач', 'Меню', and 'Шаблоны сообщений'.
- Left Panel:** A table with columns 'Id', 'Название', 'Описание', 'Тренер', and 'Программа'. It lists templates like '8 Реабилитация', '11 Кардио и Силовое', '38 Первая тренировка', and '39 новый шаблон лонг'. The '38' template is selected.
- Main Table (Упражнения):** A table with columns 'Id', 'Id упражнения', 'Название', 'Описание', 'Группа', 'Повторения', 'Продолжительн...', and 'Вес'. It lists various exercises such as '28 Приседание плie', '29 Приседания сумо', '33 Приседания с ж...', '79 "Кук-бэк"', '81 Обратные отжимания', '76 Жим штанги узким хватом', '9 Тяга в тренажере...', '126 Орбитрек', '7 Т-тяга', and '28 Приседание плie'.
- Bottom Table (Подходы):** A table with columns 'Id', 'Вес (кг)', and 'Кол-во повторений'. It shows approach details for the selected template, with weights of 38, 39, 40, and 41 kg, and 20 repetitions each.

Рисунок 32 – Форма створення шаблонів тренувань

4.15 Оцінка

В системі розроблений механізм оцінок тренерів та клієнтів. Основні типи оцінок:

- 1) клієнт оцінює тренера;
 - а) тренеру дана оцінка не відображається;
 - б) позитивна оцінка зберігається в базі;
 - в) у разі негативної оцінки для адміністратора створюється завдання для перевірки такого відгуку;
 - г) оцінити можливо один раз на тиждень з початку першого тренування;
 - д) запит для оцінки формується автоматично після заповнення всіх замірів клієнта за семиденний тренувальний цикл;
- 2) тренер оцінює клієнта;
 - а) клієнтові також не відображається оцінка тренера;
 - б) інші дії відповідають попередньому пункту.

Вибір оцінки можливий у вигляді балів (від 2 до 5) та додаванням коментаря. Всі оцінки відображені у звітах.

4.16 Відправка ваги

Клієнт має можливість відправляти свою вагу в діалог завдяки відповідному функціоналу. Здійснити це можна самостійно, зайшовши в діалог або натиснувши на задача, яка відповідає відправці ваги. В результаті кожне повідомлення такого типу розпізнається системою та даний параметр зберігається в базі даних. Кожен запис клієнта відображається на графіку змін параметра «Вага». В залежності від цих даних розраховується індикатор досягнення мети та відображається у відсотках. В адміністративному модулі налаштовується час обов'язкової відправки даного параметру. За деякий час до чергового відправлення клієнт

отримує push повідомлення на мобільний телефон з такою задачею. Це допомагає користувачу вчасно виконувати свої завдання.

4.17 Відправка прийому їжі

В мобільному додатку розроблена функціональна можливість надсилання окремого типу повідомлення, який називається «прийом їжі». Наприклад, сніданок, перекус, обід, вечеря тощо. Для кожного прийому їжі (це заздалегідь налаштовується в адміністративному модулі) створюється задача, яка має статус виконана чи ні. Відправлення прийому їжі клієнт може здійснити різними способами. Це може бути у вигляді текстового повідомлення з описанням страв, описання з прикріпленням фото або тільки фото. Фото може бути прикріплене вибравши з галереї або одразу ж камери. Тренер, зайшовши в діалог, оцінює прийом їжі клієнтом (рисунок 33).

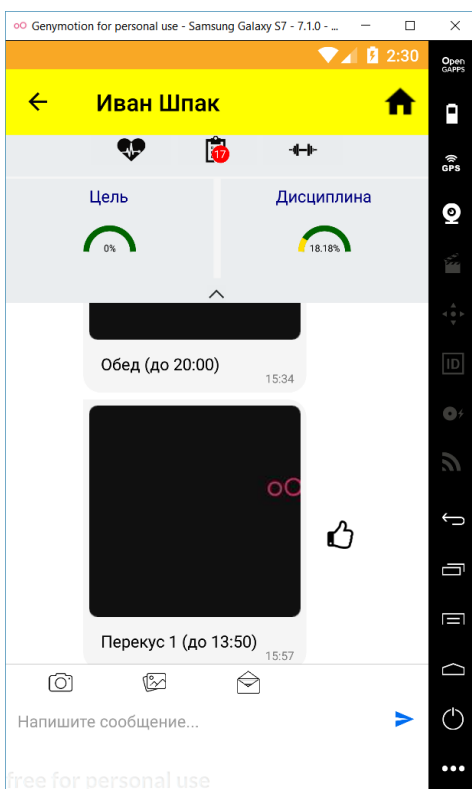


Рисунок 33 – Оцінка відправленого фото прийому їжі Клієнтом

4.18 Прикріплення фото до і після

Для тренера наявна можливість для анкети кожного клієнта додавати фото до початку тренувань і після у мобільному додатку для візуального відображення результату, який досягнув клієнт. Як і більшість, цим функціоналом передбачено редагування прикріплених фото в будь-який час в межах кожного абонементу. Вся історія прикріплених фото відображається в розрізі абонементів (згруповано) з можливістю згортати такі групи.

4.19 Завдання

Системою передбачено гнучкий механізм створення завдань для виконання кожному користувачу (рисунок 34).

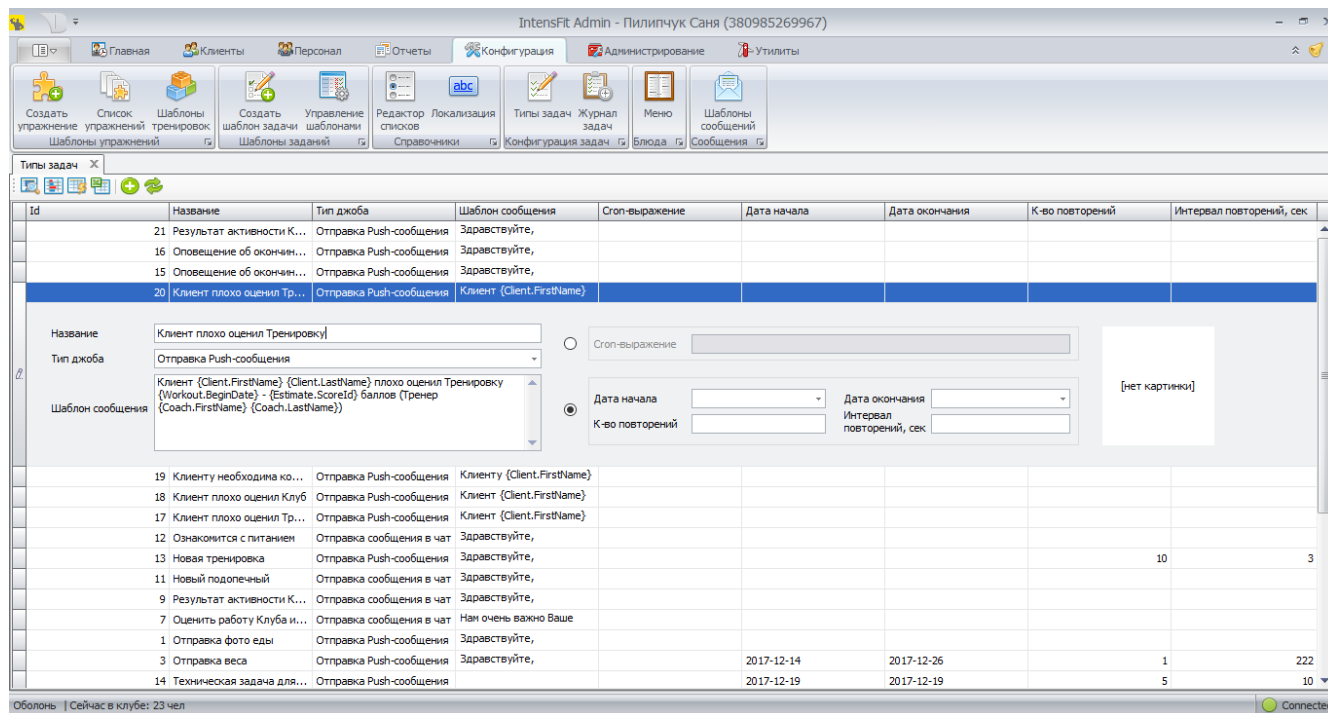


Рисунок 34 – Форма типів задач

Завдання – це сутність, яка вказує на те, що повинен виконати учасник системи. Функціонал допомагає користувачу системи розуміти, що і за чим він повинен виконувати. В адміністративному модулі є форма створення різних типів задач. Це допомагає зручно класифікувати та ідентифікувати кожну з них.

Всі завдання всіх учасників відображаються в адміністративному модулі (рисунок 35). З даної форми є можливість керувати задачами.

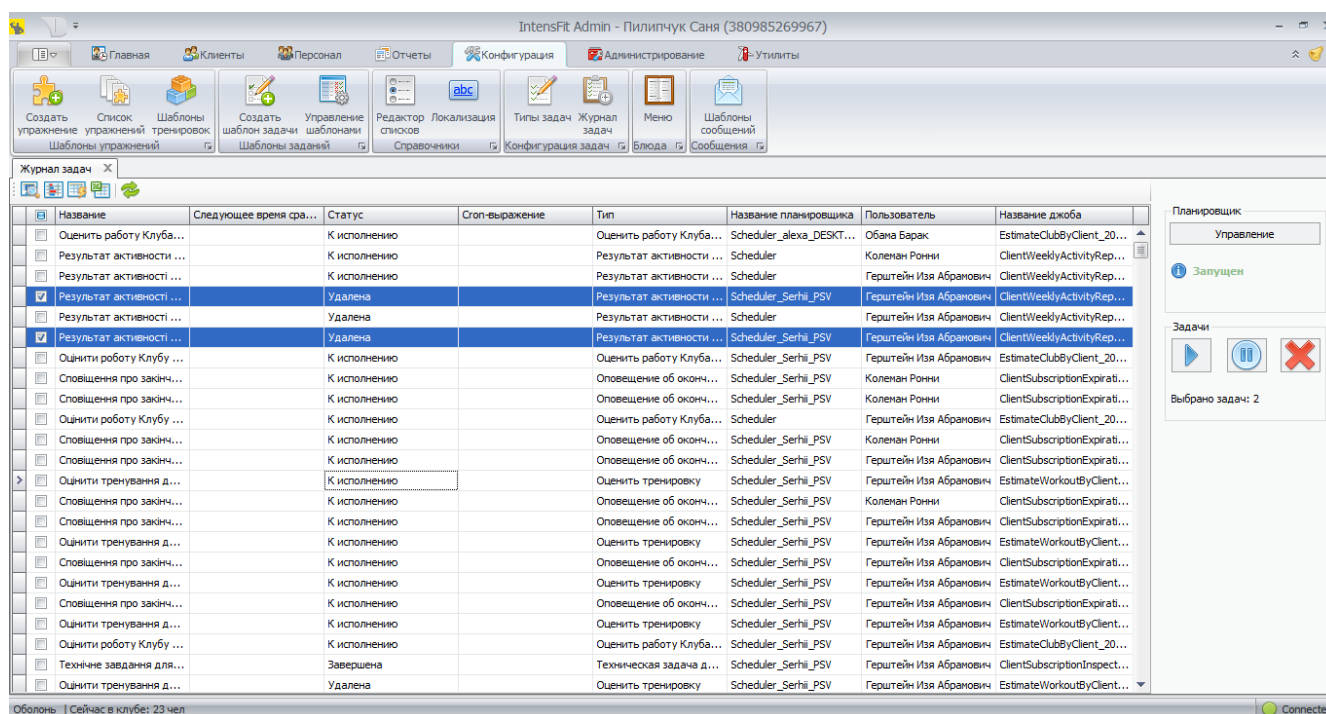


Рисунок 35 – Форма журналу задач

Завдання додаються в автоматичному режимі та для кожного користувача мобільного додатку відображаються на окремому екрані. Завдання має статус виконаної, скасованої чи простроченої. Кожне завдання логічно пов'язане з конкретним функціональним блоком системи для її виконання. При натисканні на обране завдання відкриється потрібний функціонал (екранна форма) для виконання певних дій, після чого завдання може вважатися виконаним. Кожне завдання має певний проміжок часу за який його потрібно виконати. Якщо воно не

буде виконано за відповідний час, то завдання помічається простроченим і помічається маркером.

За допомогою цього функціоналу у користувача наявна можливість зупиняти, видаляти чи запускати одну або декілька задач зі списку. На рисунку 36 зображено перелік кнопок для керування безпосередньо планувальником. Наприклад, якщо зупинити планувальник кнопкою «Зупинити», то нікому з учасників не будуть додаватися нові завдання.

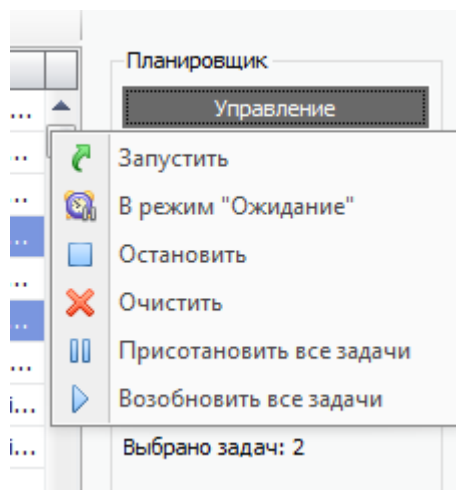


Рисунок 36 – Панель керування планувальником

Користувач в мобільному додатку отримує завдання для виконання, які відображаються на окремій формі (рисунок 37) і при виборі конкретного завдання користувач переходить безпосередньо до функціоналу (екранну форму), де він зможе виконати обране завдання.

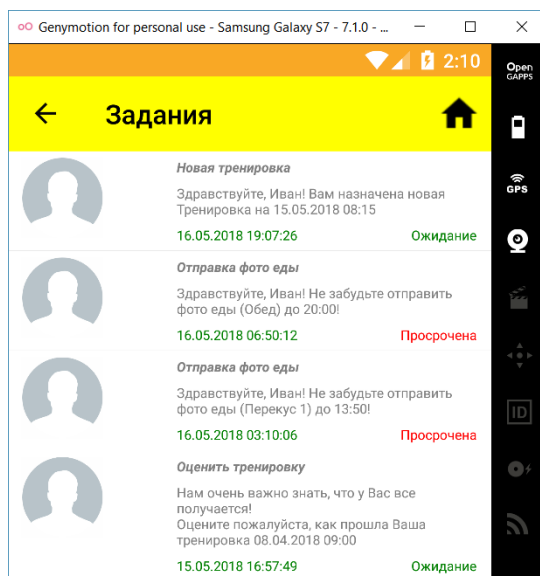


Рисунок 37 – Форма завдань

4.20 Перелік блюд

В програмному комплексі розроблено функціональна можливість додавання блюд (рисунок 38), які будуть відображатися клієнтам в мобільному додатку.

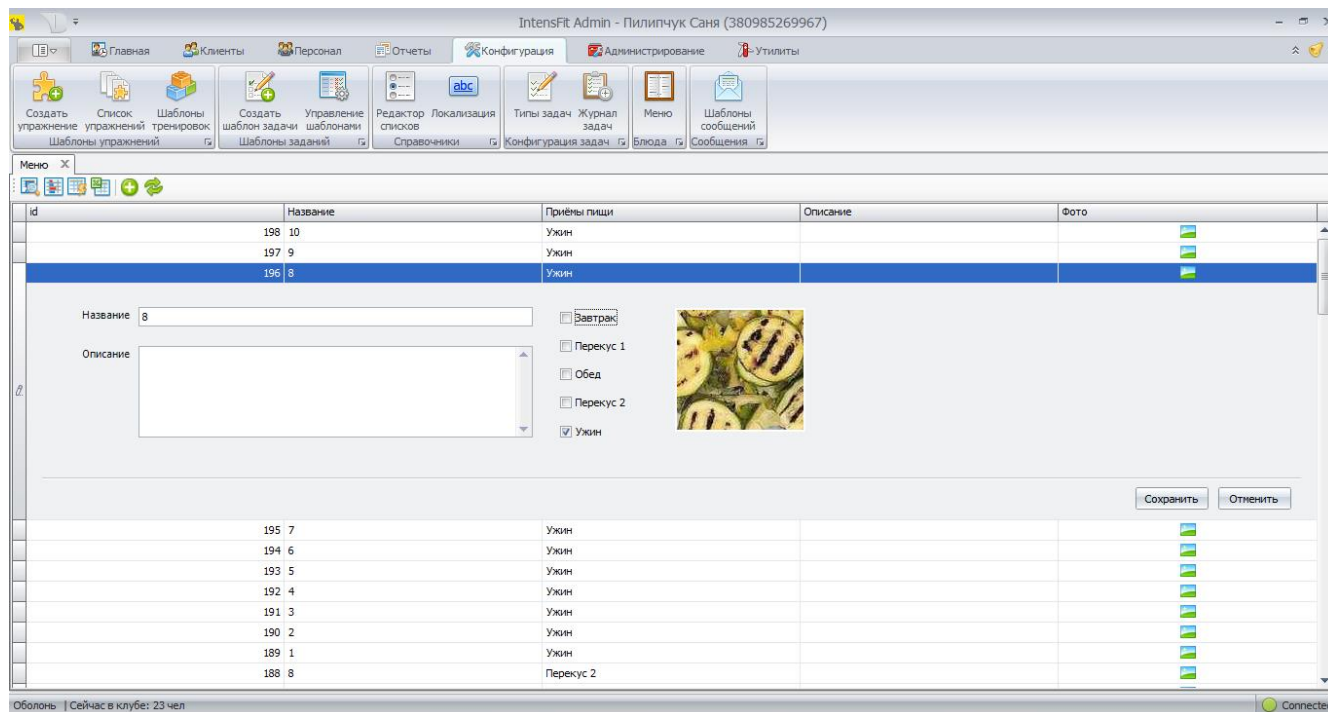


Рисунок 38 – Додавання блюд

Всі параметри для кожного блюда вказуються на формі додавання блюд в адміністративному модулі. Це допомагає клієнтові обирати певний прийом їжі, який раніше складений фахівцями.

На рисунку 39 зображена форма мобільного додатку з цим налаштованим списком. Форма має випадаючий список (налаштовується в довідниках в модулі), який дозволяє сортувати по типу прийому їжі (сніданок, обід, тощо).

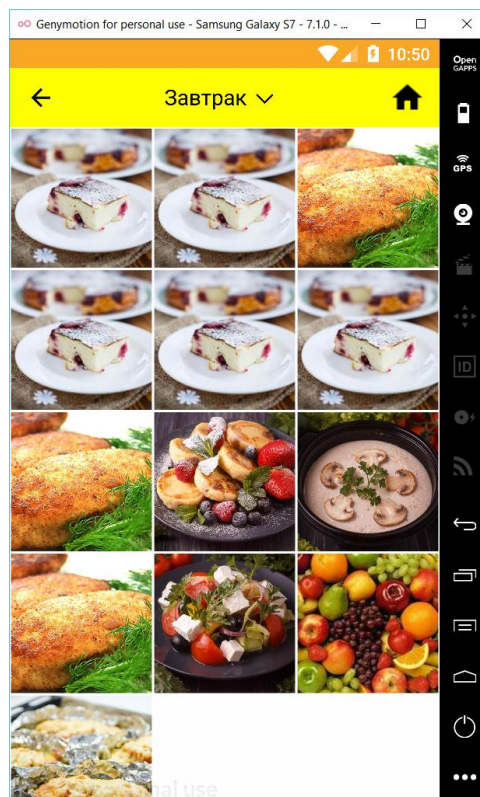


Рисунок 39 – Форма блюд в мобільному додатку

4.21 Push повідомлення

В системі налаштований механізм, який дозволяє налаштовувати та автоматично надсилати push повідомлення користувачам. Даний функціонал доступний тільки для мобільного додатку.

Push повідомлення – це короткі спливаючі повідомлення на екрані мобільного пристрою, які мають заголовок, тіло повідомлення та/або зображення. При натисканні на push повідомлення користувач може переходити в задалегідь визначену форму додатку з заданими параметрами в такому повідомленні. Такі повідомлення генеруються в залежності від завчасно визначених програмно подій в системі.

4.22 Локалізація

У системі передбачено локалізація текстових назв на трьох мовах, а саме: українська, російська та англійська. При розробці програмного комплексу задавалися ключові слова (ідентифікатори) назв кнопок, ідентифікаторів, лейблів тощо. Для кожного такого ідентифікатора додавалися переклади на трьох мовах в адміністративному модулі (рисунок 40).

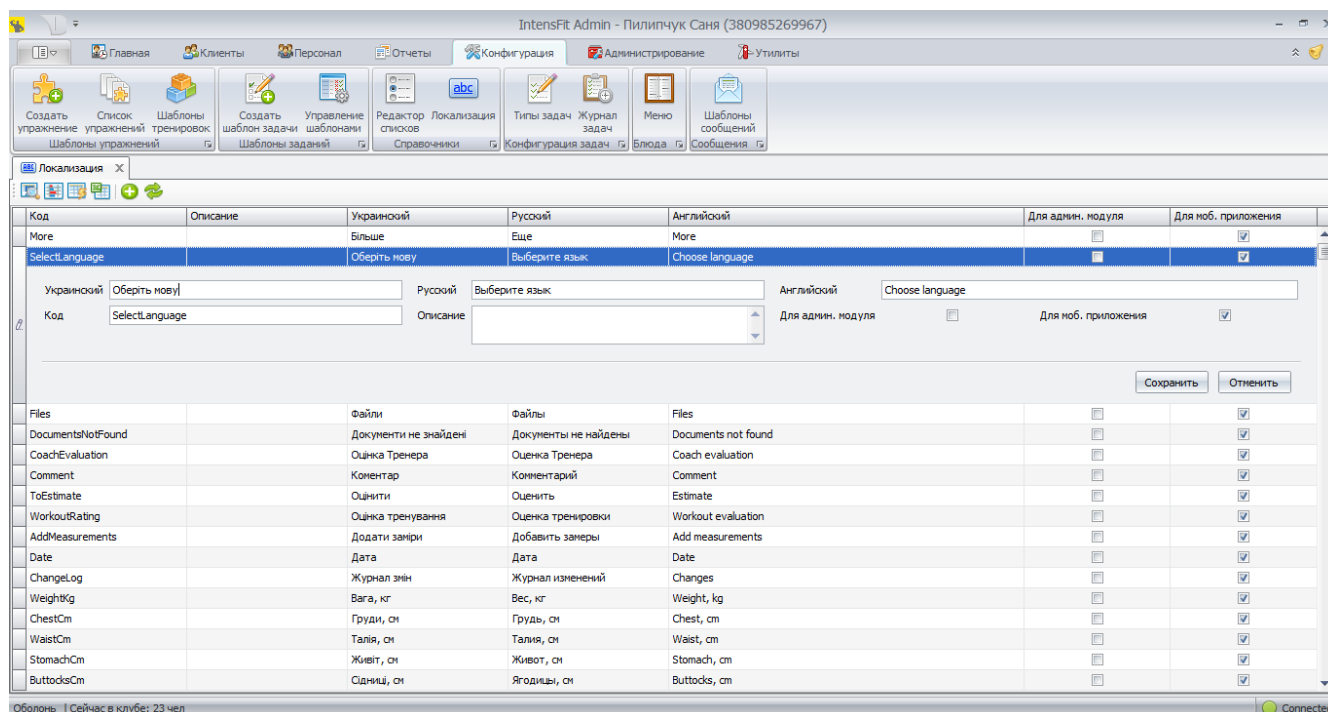


Рисунок 40 – Форма наповнення довідника локалізації

4.23 Конструктор звітів

Програмний комплекс містить конструктор звітів. Для побудови звітів використовується розроблений дизайнер FastReport (рисунок 41). За допомогою даного дизайнера можна побудувати звіти різного виду, з різним оформленням тощо. Звіти генеруються на основі даних, які містяться в базі даних. Можливо виконувати різне оформлення даних, виділяти різними кольорами, створювати таблиці, вставляти математичні формули, автоматичний розрахунок та багато іншого.

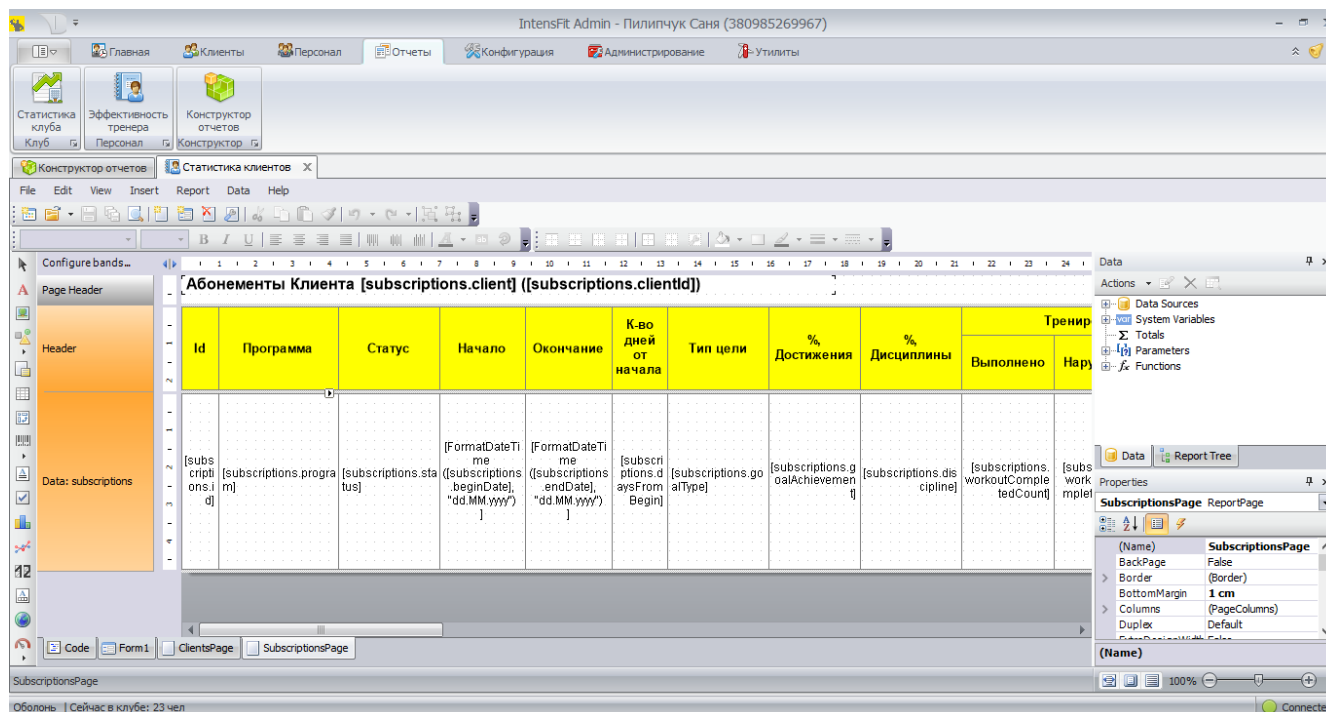


Рисунок 41 – Дизайнер для побудови звітів

4.24 Структура бази даних

Проектувавши базу даних (демонстраційний плакат 2) бралися до уваги такі поняття:

- нормалізація таблиць;

- використання індексів для підвищення швидкості виконання запитів;
- цілісність даних таблиць.

Нормальна форма – це вимога, яка пред’являється до структури таблиць в теорії баз даних для усунення з бази надлишкових функціональних залежностей між атрибутами (полями таблиць).

Нормалізація представляє процес поділу даних по окремим пов'язаним таблицями. Нормалізація усуває надмірність даних (data redundancy) і тим самим допомагає уникнути порушення цілісності даних при їх зміні, тобто уникнути аномалій зміни (update anomaly).

Як правило, нормалізація переважно застосовується при висхідному підході проектуванні бази даних, тобто коли всі атрибути, які треба зберегти в базі даних, групуємо по сутностям, для яких потім створюються таблиці. Однак при низхідному підході, коли спочатку виявляються сутності, а потім їх атрибути та зв'язки між ними, нормалізація також може застосовуватися, наприклад, для перевірки коректності спроектованих таблиць.

У ненормалізованій формі таблиця може зберігатися інформація про двох і більше сутностей. Також вона може містити повторювані стовпці або стовпці, які зберігають повторювані значення. У нормалізованій же формі кожна таблиця зберігає інформацію тільки про одну сутність.

Для прискорення процесу пошуку по таблицях, які містять велику кількість рядків (записів) служать індекси. Індокси створюються для стовпців таблиць і представлень. Індокси надають шлях для швидкого пошуку даних на основі значень в цих стовпцях. Індекс складається з набору сторінок, вузлів індексу, які організовані у вигляді збалансованого дерева. Ця структура є ієрархічною за своєю природою і починається з кореневого вузла на вершині ієрархії і кінцевих вузлів, листя, в нижній частині, як показано на рисунку 42.

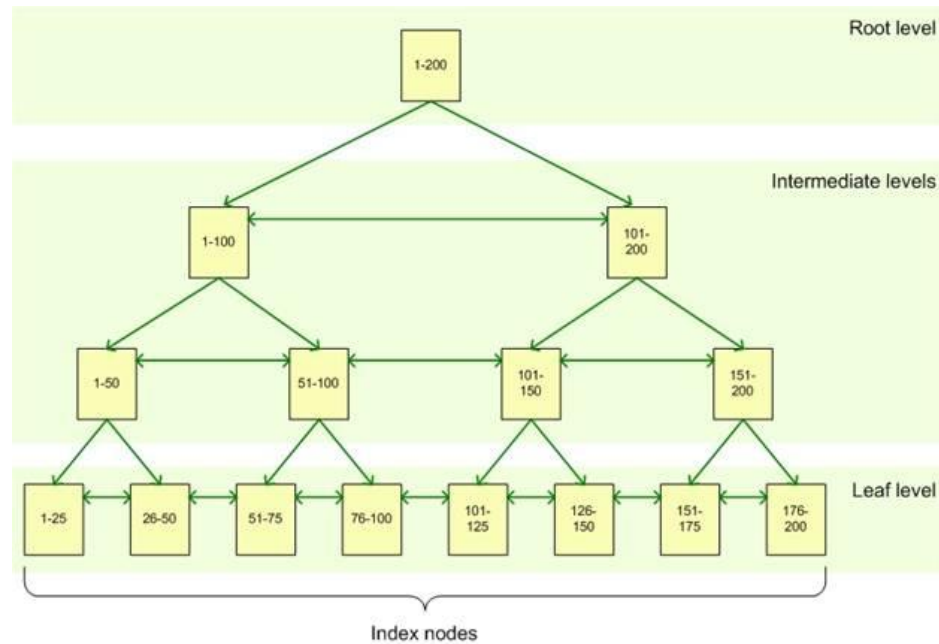


Рисунок 42 – Структура індекса

Коли формується запит на індексований стовпець, підсистема запитів починає рухатися зверху від кореневого вузла і поступово вниз через проміжні вузли, при цьому кожен шар проміжного рівня містить більш детальну інформацію про дані. Підсистема запитів продовжує рухатися по вузлах індексу до тих пір, поки не досягне нижнього рівня з листям індексу.

Індекси можуть бути або кластеризовані, або некластеризовані, можливо додаткове конфігурування складового індексу, унікального індексу або покриваючого індексу.

Висновки по розділу

В розділі описано етапи розробки алгоритмічного та програмного забезпечення програмного комплексу. Наведено архітектуру системи та взаємодію всіх її компонентів. Також описано основні функціональні частини з таблицями, зображеннями і структурою бази даних для зберігання даних користувачів. На основі даного опису формулюються вимоги для роботи програми.

5 Вимоги для роботи програмного комплексу

5.1 Вимоги до технічного забезпечення

Для адміністративного модуля на персональний комп'ютер з платформою Windows:

- процесор Pentium;
- оперативна пам'ять 512 MB;
- операційна система Windows Vista.

Для мобільного додатку (платформа Android):

- версія 4.1;
- оперативна пам'ять 512 MB.

Для мобільного додатку (платформа IOS):

- версія 9.3.

Для сервісу Web API:

- ASP.NET Web API 2 (версія 5.0.0);
- .NET 4.5;
- операційна система Windows Vista;
- IIS 7.0.

5.2 Вимоги до програмного забезпечення

- БД MS SQL 2005 або вище;
- Операційна система – Windows.

Висновки по розділу

В розділі сформульовано та описано основні вимоги до технічного та програмного забезпечення для коректної роботи системи. Для повного розуміння розробленого продукту потрібно сформулювати керівництво користувача.

6 Керівництво користувача

Управління системою відбувається за допомогою так званого адміністративного модуля, який є ключовою частиною системи. За допомогою цього модуля додаються користувачі системи, їм назначаються відповідні ролі для розподілення виконання певних дій і функцій, які доступні. Наприклад, у адміністраторів наявний повний доступ до функціональності системи. В обов'язки адміністратора входить створення користувачів, наприклад, тренерів, клієнтів тощо, призначення прав.

Створивши користувача з роллю адміністратор з'являється можливість додавати анкети користувачів з роллю тренер та клієнт. Перш за все створюються тренери, яким вказується потрібний тип і контактна інформація. Головна мета таких користувачів – це взаємодія з клієнтами, керування абонементом, створення тренувань, контроль виконання певних завдань клієнтами тощо.

Для того, щоб почати працювати з клієнтом потрібно створити його анкету в системі. Анкета створюється за допомогою форми створення в адміністративному модулі. В профілі обов'язкове заповнення деяких параметрів, таких як блок питань про здоров'я. Для прикріплення документів клієнта створена окрема форма «Документи». Можуть бути прикріплені довідки про стан здоров'я чи рекомендації з приводу харчування тощо.

Наступним кроком потрібно виконати активацію клієнта, яку може виконати користувач з відповідними ролями в системі.

Наступний крок, який повинен бути виконаний користувачем з відповідними ролями в системі – активувати профіль клієнта. Згідно процесу у системі, це виконується шляхом створення абонементу для раніше зареєстрованого клієнта. Як було зазначено вище, абонемент містить випадючий список, де потрібно вказати тренера, який співпрацювати з клієнтом до закінчення строку абонементу та досягненню бажаного результату. Під час дії абонементу є

можливість змінювати тренера для клієнта (наприклад, в зв'язку з хворобою). Після виконаних вище зазначених дій автоматично створюється діалог з обраним тренером. Діалог пришвидшує комунікацію та виконує ряд стратегічно важливих функцій (відправка ваги, прийому їжу, фото тощо).

У адміністративному модулі є можливість редагувати профілі користувачів та інших сутностей за наявності відповідних ролей.

У десктопному додатку користувач може наповнювати або змінювати такі довідники:

- шаблони повідомлень – шаблони повідомлень з параметрами, які доступні тренеру з діалогу в мобільному додатку;
- вправи – перелік фізичних вправ. Довідник використовується, як один із параметрів при створенні тренування;
- універсальний довідник – складається з простих списків для вибору в випадуючих списках системи. Наприклад, стать користувача, країна тощо;
- локалізація – довідник, у якому зберігаються переклади різних назв частин системи. Доступно на мовах українська, російська та англійська.
- типи завдань – слугують для класифікації завдань. Створюючи завдання вказується його тип.
- меню страв – використовується для формування різних типів прийомів їжі клієнтом;

Після того, як було налаштовано профіль користувача в адміністративному модулі, він має доступ до мобільного додатку. Користувач проходить автентифікацію та авторизацію за допомогою введення мобільного номеру телефону та пароль. При першому вході обов'язкова авторизація, всі наступні – не потребують такої процедури. При вході аналізується тип користувача та розмежовується функціонал в мобільному додатку. Розглянемо дві основні ролі системи.

6.1 Тренер

Користувач з даною роллю має головну форму, яка відображається одразу після авторизації, яка містить список діалогів з призначеними йому клієнтами, головну панель у верхній частині екрану. На цій панелі розміщені кнопки з переходами на сторінку з учасниками, які мають абонемент з типом онлайн та на сторінку зі своїми поточними завданнями для виконання. Також під цією панеллю з кнопками розміщений рядок пошуку діалогу по назві. Це зроблено для зручності, так як у тренера набагато більше діалогів в порівнянні з клієнтом. Кожен діалог на списку діалогів містить фотографію клієнта, назву діалогу (в даному випадку це ім'я клієнта) та останнє відправлене повідомлення. Натиснувши на фото клієнта здійснюється перехід на профіль клієнта. Дану маніпуляцію можна здійснити перейшовши в діалог і натиснувши на фото клієнта на відправленому їм повідомленні.

В анкеті клієнта можна переглядати інформацію або керувати основними сутностями, такими як заміри, тиск і пульс, фото клієнта до початку занять і після, оцінка клієнта та безпосередньо тренування. Для абонементу, який створений в адміністративному модулі, тренер додає тренування. Тренування складається з набору вправ та підходів, ваги, часу виконання тощо для кожної вправи. Після деякого проміжку часу та виконанням певної кількості тренувань, тренер оцінює клієнта та додати коментар.

У діалозі тренера з клієнтом наявна верхня панель, на якій він також може перейти в анкету клієнта, створені тренування та актуальні задачі.

Внизу сторінки діалогу тренер має поле введення повідомлення, декілька функціональних кнопок: ставка в поле вводу повідомлення шаблону повідомлень, додавання фото з камери або галереї та кнопка відправки повідомлення.

Повідомлення, які відправляє клієнт розрізняються по типу. Звичайне текстове повідомлення та/або з фото (зображенням), повідомлення типу «Прийом

їжі» і «Вага». Повідомлення з типом «Прийом їжі» тренер оцінює відповідною позначкою біля повідомлення. Оцінка може бути таких варіантів як: задовільно, незадовільно і не оцінено. Цей функціонал служить для контролю правильного харчування клієнта.

6.2 Клієнт

Ввійшовши в мобільний додаток клієнт має відмінний набір функціоналу від користувача з роллю тренер:

- панель, яка розміщена вверху екрану, містить кнопки переходу на свій профіль, актуальний список завдань, сторінку «Здоров'я» (додається при створенні анкети), тренування (в рамках конкретного абонементу) та сторінку з вибором рекомендованих та згрупованих по прийомам їжі страв. На сторінці з тренуваннями, Клієнт має можливість розпочати тренування, відмічати виконання вправ (проставляючи відсоток виконання від 25% до 100%) та завершати тренування вказавши коментар (не обов'язково);

- нижче два великих індикатори с відображенням шкали і відсотків. Натиснувши на відповідний відбувається перехід на сторінки перегляду цілі та дисципліни клієнта. На них же відображається історія початкових, кінцевих та проміжних результатів, які досяг клієнт та їх візуалізація на графіках;

- наступним відображається список діалогів. Діалог виглядає аналогічно сторінці діалогів для тренера. Верхня панель відмінна та повторює панель із попереднього екрану. Форма також містить кнопки додавання прийому їжі, відправки ваги тренеру, прикріплення фото з галереї або з камери, кнопка відправки повідомлення.

Клієнт може переходити на анкету тренера, натиснувши на фото або ім'я біля відправленого повідомлення (аналогічно функціоналу по переходу в анкети клієнта або тренера, який описаний вище). Клієнт має можливість оцінити тренера

та додати коментар до оцінки. Додані тренером фото до і після клієнт має можливості тільки переглядати.

6.3 Загальна інформація

В процесі користування мобільним додатком і виконанням певних дій та процесів додаються завдання для виконання кожним користувачем. Завдання можуть призначатися з обмеженим часом виконання, з точно вказаним часом або необмежені в часі.

Всі користувачі мобільного додатку на головному екрані мають кнопку «Ще» для переходу в налаштування додатку, зміни мови інтерфейсу, правил клубу та кнопку виходу з облікового запису.

Висновки по розділу

У розділі розглянуто керівництво для цільового користувача системою, послідовність дій для більш швидкого входження в систему та розуміння можливостей розробленого продукту.

7 Маркетинговий аналіз стартап-проекту

7.1 Опис ідеї проекту

Таблиця 7.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Програмний комплекс «IntensFit» – це проста та, головне, зручна у використанні система для організації обліку і контролю клієнтів в тренажерному залі, а також мобільний додаток для полегшення спілкування і співпрацювання клієнта з тренером для досягнення поставлених цілей.	Тренажерні зали, клуби або дистанційні заняття з клієнтом (програма «Інтенсив онлайн» та «Інтенсив харчування»)	Поліпшення якості, швидкості та зручності обслуговування клієнтів в сфері послуг.

Таблиця 7.2 – Опис ідеї стартап-проекту

№	Техніко-економічні характеристики ідеї	Продукція конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Отмечалка	Fitness Pro				
1	Рольова модель доступу	+	+	-			+	
2	Медичні довідки та абонемент	+	+	+			+	
3	Профіль клієнта	+	+	+			+	
4	База клієнтів	+	+	+			+	
5	Звіти	+	-	+		+		
6	Використання для декількох клубів	+	-	+		+		

Закінчення таблиці 7.2

7	Мобільний додаток Android	+	+	-				+
8	Мобільний додаток IOS	+	-	-				+
9	Локалізація	+	-	-			+	
10	Шаблони блюд	+	-	-				+
11	Шаблони повідомлень	+	-	-				+
12	Шаблони тренувань	+	-	-				+
13	Задачі	+	-	-				+
14	Push повідомлення	+	-	-				+
15	Діалоги	+	-	-				+

7.2 Технологічний аудит ідеї проекту

Таблиця 7.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Адміністративний модуль	DevExpress	Наявна	Платно для корпоративної розробки
2	Веб-сервіс	.NET Web API	Наявна	Доступна безкоштовно
3	Мобільні додатки	.NET Xamarin	Наявна	Доступна безкоштовно
4	База даних	MS SQL	Наявна	Доступна безкоштовно
5	Розміщення в хмарному середовищі	Microsoft Azure	Наявна	Платно

7.3 Аналіз ринкових можливостей запуску стартап-проекту

Таблиця 7.4 – Попередня характеристика потенційного ринку

№	Показники стану ринку	Характеристика
1	Кількість головних гравців, од	2
2	Загальний обсяг продаж, грн./ум.од	360 тис. грн на рік
3	Динаміка ринку	Повільно, але зростає
4	Наявність обмежень для входу	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Придбання ліцензії для технології DevExpress
6	Середня норма рентабельності в галузі або по ринку, %	60%

Висновок: враховуючи кількість головних гравців по ринку, зростаючу динаміку ринку, невелику кількість конкурентів та середню норму рентабельності можна зробити висновок, що на даний момент, ринок для входження стартап-продукту є привабливим.

Таблиця 7.5 – Характеристика потенційних клієнтів стартап-проекту

№	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці цільових груп клієнтів	Вимоги споживачів до товару
1	Обслуговування клієнтів	Працівники та відвідувачі тренажерного клубу	Потреби в різній функціональності	Ознайомитися з системою та, можливо, запросити особливі налаштування системи
2	Швидкість обслуговування	Працівники та відвідувачі тренажерного клубу	Потреби в різній функціональності	

Закінчення таблиці 7.5

3	Якість обслуговування	Працівники та відвідувачі тренажерного клубу	Потреби в різній функціональності	
---	-----------------------	--	-----------------------------------	--

Таблиця 7.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Конкуренти	Наявність конкурентів котрі надають схожі рішення	Зменшення ціни на поставлену послугу; Розробка унікальних характеристик товару; Надання ліцензій на обслуговування
2	Кошти на розробку та підтримку продукту	Закінчення грошей та недостатнє фінансування	Залучення додаткових інвесторів, мотивація роботи на перспективу; Ітеративна розробка продукту задля покрокового виведення продукту на ринок та отримання відповіді користувачів
3	Вихід аналогу	Вихід аналогу даного товару може призвести до знецінення та безідейності даного товару	Вихід товару на ринок в коротші строки з не повною, але достатньою, функціональністю для зацікавлення усіх цільових аудиторій; Проведення рекламної компанії

Таблиця 7.7 – Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Новий продукт	Вихід на ринок, Зменшення монополії, Надання нових рішень у сфері	Розробка нової функціональності; Вихід нової продукції на ринок; Надання різноманітних типів ліцензій в залежності від потреб користувача \ замовника.
2	Вихід аналогу	Надати продукт з певними характеристиками та можливостями що відсутні у компаній конкурентів	Аналіз ринку та користувачів задля задоволення їх потреб та надання функціональності у найкоротші строки за ціну, котра є дешевшою ніж у продуктів-замінників.

Закінчення таблиці 7.7

3	Зворотній зв'язок від користувачів	Можливість отримання необхідної інформації для вдосконалення продукту	Наявність вхідних даних та реакція на них з боку команди розробників задля задоволення потреб та бажань кінцевих користувачів системи кешування даних.
4	Грошова винагорода за рекламу	При достатньому попиту на систему кешування даних можлива комерціалізація продукту на основі реклами задля отримання грошової винагороди для подальшого розвитку продукту та оплати заробітної плати працівникам	Точкова комерціалізація продукту; Введення реклами; Ведення додаткових коштів у проект задля його подальшого розвитку.

Таблиця 7.8 – Ступеневий аналіз конкуренції на ринку

№	Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1	Тип конкуренції: монополістична	Товар від кожної компанії на ринку, являється недосконалим замінником товару, реалізованого іншими фірмами; На ринку є умови для входу та виходу; Ціна корелює між суперниками;	Розробка продукту з характеристиками, які покривають сфери вживання що не покривають інші товари-замінники; Кореляція цін у відповідності до товарів замінників; Різні типи ліцензій.
2	Рівень конкурентної боротьби: світовий	Всі продукти замінники розроблялись інтернаціональними командами з усього світу, продукти не належать до певної держави, а належать команді розробників	Вихід на ринок збуту продукту з клієнто-необхідною функціональністю; Налагодження маркетингу на основних Інтернет ресурсах задля охоплення великої кількості потенційних користувачів; Надання бета-версій продукту.

Закінчення таблиці 7.8

3	Галузева ознака: внутрішньогалузева	Даний тип продукту може використовуватися тільки у сфері розробки ІТ додатків \ продуктів	Надання зручного, інтуїтивно зрозумілого інтерфейсу; Підтримка всім відомих методів взаємодії з середовищем розробки; Наявність документації та он-лайн підтримки.
4	Конкуренція за видами товарів: товарно-видова	Дана конкуренція – конкуренція між товарами одного виду.	Впровадження функціональності яка відсутня у товарів-замінників; Спрощення інтерфейсів; Надання підтримки.
5	Характер конкурентних переваг: цінова та не цінова	Цінові переваги – точкова комерціалізація; Не цінова – надання функціональності, що відсутня у товарах-замінниках.	Надання платних ліцензій лише на критично важливу функціональність для клієнта з певним строком підтримки, що зазначена у відповідній ліцензії; Впровадження унікальної функціональності.
6	За інтенсивністю: марочна	Наявність унікального знаку що відрізняє даний продукт від продуктів-замінників	Впровадження власної назви та власного знаку.

Таблиця 7.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Середня	Низька	Низька	Середня	Висока
Висновки	На даний час конкурентів в даній області невелика кількість в порівнянні з розмірами ринку клієнтів для системи	Можлива поява нових подібних розробок для тренажерних залів, але прогрес у цьому напрямку не дуже стрімкий	Немає залежності від постачальників, так як використовується програмне забезпечення, яке розроблене власноруч без додаткових постачачів від третіх осіб	Клієнти уже можуть використовувати системи, які їх влаштовують і продати свою систему стає складніше. Може бути перешкодою їхнє небажання змінювати внутрішні процеси	Висока імовірність появи нових систем або замовлення чи покупка подібних систем із-за кордону, які можуть замінити основні процеси та задовільнити потреби

Проаналізувавши можливості роботи на ринку з огляду на конкурентну ситуацію можна зробити висновок: оскільки кожний з існуючих продуктів не впливає у великій мірі на поточну ситуацію на ринку в цілому, кожний з існуючих продуктів має свою специфічну сферу використання та свої позитивні та негативні сторони щодо рішення певних типів задач, то робота та вихід на даний ринок є можливою і реалізованою задачею.

Для виходу на ринок продукт повинен мати функціонал що відсутній у продуктів-аналогів, повинен задовольняти потреби користувачів, мати необхідний та достатній функціонал з конфігурування, підтримку зі сторони розробників та можливість розробки спеціального функціоналу за відповідною ліцензією.

Таблиця 7.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування
1	Рольова модель	Розподілення функцій в залежності ролі користувача
2	Звіти	Підготовка звітності для аналізу ефективності роботи та різних функцій
3	Використання в декількох клубах	Можливість масштабування
4	Мобільні додатки на Android і IOS	Зручність та мобільність використання системи
5	Локалізація	Можливість залучення англомовних та російськомовних клієнтів
6	Шаблони блюд	Підготовлені набори блюд для правильного харчування
7	Шаблони повідомлень	Більша швидкість відповіді клієнтам
8	Шаблони тренувань	Швидкість підготовки тренувань клієнтам та зменшення втрати часу на це тренеру
9	Задачі	Розуміння користувачів, які задачі і коли потрібно виконувати
10	Push повідомлення	Зручність та швидкість реагування на події

Таблиця 7.11 – Порівняльний аналіз сильних та слабких сторін системи

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з запропонованим						
			-3	-2	-1	0	+1	+2	+3
1	Рольова модель					X			
2	Звіти				X				
3	Використання в декількох клубах					X			
4	Мобільні додатки на Android і IOS		X						
5	Локалізація			X					
6	Шаблони блюд		X						
7	Шаблони повідомлень		X						
8	Шаблони тренувань		X						
9	Задачі			X					
10	Push повідомлення					X			

Таблиця 7.12 – SWOT аналіз стартап-проекту

<p>Сильні сторони (S):</p> <ul style="list-style-type: none"> – Актуальні технологічні рішення – Можливість масштабування – Компонентна архітектура – Мобільні додатки – Багатий та легко розширюваний функціонал адміністративного модуля – Підтримка – Розробка нового функціоналу по запити та без 	<p>Слабкі сторони (W):</p> <ul style="list-style-type: none"> – Адміністративний модуль – це десктопний додаток для Windows – Потребує певну кількість затрат часу розробника – рефакторингу окремих частин системи – Немає веб-версії модуля
---	---

Закінчення таблиці 7.12

Можливості (О):	Загрози (Т):
<ul style="list-style-type: none"> – Залучення нових людей в команду для розширення проектної команди – Пошук нових клієнтів для системи – Ребрендинг системи та налаштування під інших клієнтів – Доопрацювання мобільних додатків та виділення їх як окремих незалежних частин для інших потреб ринку 	<ul style="list-style-type: none"> – Поява аналогічної системи з більшим набором функціоналу – Аналогічна система з веб-версією – Якість обслуговування конкурентної системи вища – Нестача ресурсів, часу та коштів на розвиток системи

Таблиця 7.13 – Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Безкоштовне надання певного функціоналу у користування споживачам на обмежений термін	Головний ресурс – люди, даний ресурс - наявний	2-3 місяці
2	Реклама	Залучення власних коштів для реклами товару	1-2 місяці
3	Написання статей та опис товару на відомих ресурсах	Головний ресурс – час, даний ресурс - наявний	2-3 тижні
4	Презентація товару на хакатонах й інших ІТ заходах	Ресурс – час та гроші для участі, наявні	1-3 місяці

7.4 Розроблення ринкової стратегії проекту

Таблиця 7.14 – Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Працівники тренажерних клубів та клієнти цих клубів	Система не складна для розуміння, тому поріг входження низький	Попит достатній	Конкуренція зростає, тому що популярність занять спортом зростає	На даний момент цілком можливо
Які цільові групи обрано: працівники та відвідувачі тренажерних клубів					

Відповідно до проведеного аналізу можна зробити висновок, що підходящою цільовою групою для розповсюдження даного програмного продукту є працівники ІТ сфери, ІТ компанії в цілому та будь-які підприємства котрі використовують програмні продукти побудовані на мові програмування Java, та використовують реляційні бази даних. Відповідно до стратегії охоплення ринку збуту товару обрано стратегію масового маркетингу, оскільки для підприємств, ІТ працівників та ІТ компаній у цілому надається стандартизований продукт з можливістю розширення функціональності за домовленістю (відповідно до ліцензії).

Таблиця 7.15 – Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Надання функціональності що відсутня у товарів-замінників, підтримка клієнтів	Проведення реклами, освітлення унікальної функціональності через інтернет ресурси та інші канали, контакт напряду з споживачами; формування лояльності і прихильності споживачів	Зниження ступеню замінності товару; Прихильність клієнтів; Відмітні властивості товару; Відмітні характеристики товару;	Стратегія диференціації

Таблиця 7.16 – Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, які?	Стратегія конкурентної поведінки
Ні, оскільки є товари-замінники, але дані товари замінники не мають деякого необхідного функціоналу	Так, ціль компанії знайти нових споживачів та, частково, забрати існуючих у конкурентів задля задоволення потреб останніх	Компанія частково копіює характеристики товару конкурента, основна ціль компанії розробка нового унікального функціоналу, з підтримкою основного функціоналу конкурентів	Стратегія заняття конкурентної ніші

Таблиця 7.17 – Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
1	Зручність у використанні	Аналіз бажань користувачів	Інтуїтивно зрозумілий інтерфейс	Зручність

Закінчення таблиці 7.17

2	Наявність функціоналу для полегшення співпраці з клієнтом	Планування, розробка та тестування	Наявність діалогів, Push повідомлення, додавання завдань	Швидкість комунікації та зворотнього зв'язку
3	Тренування, харчування, діалоги та звітність	Навчання персоналу клубу використання даного функціоналу	Створення тренувань, додавання блюд та побудова звітів	Зворотній зв'язок та візуальне бачення результатів

Відповідно до проведеного аналізу можна зробити висновок, що стартап-компанія вибирає як базову стратегію розвитку – стратегію диференціації, як базову стратегію конкурентної поведінки – стратегію заняття конкурентної ніші.

7.5 Розроблення маркетингової програми стартап-проекту

Таблиця 7.18 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Швидкість	Функції для автоматизації процесів	Діалоги, Push повідомлення, тренування, вправи, звіти тощо
2	Якість	Безперервна робота системи з підтримкою	Зворотній зв'язок, допомога у вирішенні питань
3	Зручність		Інтуїтивно зрозумілий продукт

Таблиця 7.19 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
1. Товар за задумом	Автоматизована система обслуговування відвідувачів тренажерного залу на платформі .NET, мова програмування C#		
2. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх/Тл/Е/Ор
	Автоматизація надання послуг	Нм	Вр/Е/Ор

Закінчення таблиці 7.19

	Зменшення витрати часу на обслуговування одної особи	Нм	Вр/Е/Ор
	Зручність для клієнтів – мобільний додаток	Нм	Вр/Тх/Тл
	Адміністративний модуль для керування клубом	Нм	Вр/Тх/Тл
	Компонентна розробка	М	Тх
	Автоматизоване генерування завдань	Нм	Вр/Тх
	Якість: дотримання стандартів, нормативів та розробка кейсів для тестування тощо		
	Назва організації-розробника: «NinePlanet», назва товару: «IntensFit»		
	Ліцензія на використання		
3. Товар із підкріпленням	До продажу: наявна повна документація, акції на придбання декількох ліцензій, знижки для певних сегментів на покупку товару		
	Після продажу: додаткова підтримка спеціалістів, налаштування, підтримка збоку розробника		
За рахунок чого потенційний товар буде захищено від копіювання: захист інтелектуальної власності, патент			

Таблиця 7.20 – Визначення меж встановлення ціни

Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
2 600 грн/міс	2 800 грн/міс	> 50 000 грн/міс	2 000 грн/міс – 5 000 грн/міс

Таблиця 7.21 – Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Продукти індивідуальних осіб або компаній	Надання консультації та ліцензій, встановлення ціни, акцій	Збут від імені компанії власною командою	Збут через рекламування та пошук індивідуальних клієнтів з бажанням розвитку проведення індивідуальних тренувань

Таблиця 7.22 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Пошук відповідних компаній	Знайомства та рекомендації, Інтернет	Якість продукту та набір функціоналу	Висвітлення ідеї продукту та гнучкості	Коротке описання функціоналу, акцент на мобільному додатку та зручній взаємодії з клієнтом тощо
2	Пошук індивідуальних продуктів		Надійність та підтримка	Презентація рішення та гарантія	

Як результат було створено ринкову (маркетингову) програму, що включає в себе визначення ключових переваг концепції потенційного товару, опис моделі товару, визначення меж встановлення ціни, формування системи збуту та концепцію маркетингових комунікацій.

Висновки по розділу

В четвертому розділі описано стратегії та підходи з розроблення стартап-проекту, визначено наявність попиту, динаміку та рентабельність роботи ринку, як висновок було вказано що існує можливість ринкової комерціалізації проекту. Розглянувши потенційні групи клієнтів, бар'єри входження, стан конкуренції та конкурентоспроможність проекту було встановлено що проект є перспективним. Розглянуто та вибрано альтернативу впровадження стартап-проекту та доведено доцільність подальшої імплементації проекту.

Висновки

У дипломному проекті було розглянуто програмну систему «IntensFit» для тренажерних залів. Дана система має широкий набір функціоналу і включає дві основні частини – адміністративний модуль та мобільний додаток. Весь обмін інформацією та робота з даними відбувається за допомогою веб-сервісу. Обрана та побудована архітектура дозволяє легко та зручно контролювати (додавати, видаляти або переміщати) елементи системи, що є вагомим фактором для подальшого її розвитку та підтримки.

Продукт робить зручним та швидким обслуговування клієнтів в даній сфері життєдіяльності. Основна програма тренувань в системі включає в себе:

- відвідування тренажерного залу;
- персональні тренування з тренером;
- складання програми харчування;
- щотижневий контроль результатів.

Система зберігає всі завдання, результати до, під час і після занять, що дає змогу легко відслідкувати динаміку зміни результатів та, якщо потрібно, наприклад, змінити навантаження, раціон харчування тощо. Завдяки цьому взаємодія Тренера і Клієнта стає максимально простою, невимушеною та, головне, сприяє швидкому досягненню бажаних результатів за короткий проміжок часу.

Розробка виконувалась на основі сервісно-орієнтованого підходу. Веб-сервіс розроблений за допомогою технології ASP.NET Web API, яка входить до платформи .NET і підтримує протокол REST. ASP.NET – це технологія створення веб-застосунків і веб-сервісів від компанії Майкрософт. ASP.NET має перевагу у швидкості в порівнянні з іншими технологіями, заснованими на скриптах (PHP, тощо). Розширюваний набір елементів управління і бібліотек класів дозволяє швидко розробляти застосунки.

Адміністративний модуль розроблений з використанням DevExpress Winforms. Ця технологія являє собою програмний пакет, в складі якого знаходиться велика кількість елементів керування і бібліотек, які розширюють наявний функціонал Windows Forms. Завдяки використанню даної технології, швидко розробляти гнучкі налаштовувані додатки зручно та швидко.

Мобільний додаток виконано за допомогою зручної технології Xamarin Froms, яка також входить до складу платформи .NET і дозволяє розробляти мобільні додатки відразу на декілька платформ (наприклад, Android і IOS).

Обрані технології є сучасними та перспективними, що дасть змогу розвивати удосконалювати та розвивати систему. Планується розробити:

- функціонал автоматичної звітності для швидкого збору результатів роботи користувачів та системи;
- СМС повідомлення, які можна відправляти з системи окремому користувачу або ж групі користувачів;
- можливість інтегрування систему з карточною системою допуску в тренажерний зал. Перед входом в тренажерний зал Клієнт прикладає ключ-карту до зчитувального пристрою і за допомогою внутрішньої системи контролю доступу фіксується відвідування. З метою зручного відображення в мобільному додатку для Тренера інформації, які Клієнти в даний момент знаходяться в тренажерному залі.

Перелік посилань

1. Рихтер, Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Дж. Рихтер — К.: ООО Издательство «Питер», 2013. — 896 с.
2. Троелсен, А. Мова програмування C# 5.0 і платформа .NET 4.5 / Дж. Рихтер — К.: ООО «И.Д. Вильямс», 2013. — 1310 с.
3. Мартін Р. С., Мартин М. Принципи, патерни і методики гнучкої розробки на мові C #; Символ-Плюс, 2011. - 768 с.
4. Фаулер, М. Улучшение существующего кода / М. Фаулер — К.: ООО Издательство «Символ-Плюс», 2013. — 708 с.
5. Мартин, Гр. Введение в SQL / Гр. Мартин – К.: Лори, 2010. – 227 с.
6. Буч, Гр. Язык UML. Руководство пользователя / Гр. Буч. – Москва: ДМК Пресс, 2007. – 200 с.
7. Лайза Криспин, Джанет Грегори Гибкое тестирование: практическое руководство для тестировщиков ПО и гибких команд = Agile Testing: A Practical Guide for Testers and Agile Teams. — М. : «Вильямс», 2010. — 464 с.
8. Калбертсон Роберт, Браун Крис, Кобб Гэри Быстрое тестирование. — М. : «Вильямс», 2002. — 374 с.
9. Бейзер Б. Тестирование чёрного ящика. Технологии функционального тестирования программного обеспечения и систем. — СПб. : Питер, 2004. — 320 с.
10. Отмечалка [Електронний ресурс] : [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <https://otmechalka.com/>
11. FitnessPro [Електронний ресурс] : [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <http://fitness.aihelps.com/>
12. Microsoft Developer Network [Електронний ресурс] : [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <https://msdn.microsoft.com/>

- 13.Справочний центр системи Отмечалка [Електронний ресурс] : [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <https://wiki.otmechalka.com/>
- 14.SlideShare [Електронний ресурс]: [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <https://www.slideshare.net/genocyber/aspnet-5-47551314>
- 15.Quora [Електронний ресурс]: [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <https://www.quora.com/NET-Framework-How-do-you-implement-the-model-view-controller-MVC-pattern-for-a-desktop-application-in-C>
- 16.Daily .NET Tips [Електронний ресурс]: [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <https://dailydotnettips.com/understanding-the-navigation-pattern-for-ios-mobile-app-development/>
- 17.Merle-amber blogspot [Електронний ресурс]: [Веб-сайт]. – Електронні дані – [Київ]. – Режим доступу <http://merle-amber.blogspot.com/2012/11/real-time-aspnet-signalr.html>
- 18.Fitness aihelps [Електронний ресурс]: [Веб-сайт]. – Електронні дані– [Київ]. – Режим доступу <http://fitness.aihelps.com/>

Додаток А
Демонстраційні плакати

Додаток Б
Результат перевірки роботи на співпадіння



Submission author:
Лісовиченко Олег Іванович ред
Check date:
27.11.2019 06:28:08 GMT+0
Report date:
27.11.2019 06:29:16 GMT+0

Check ID:
1000652556

Check type:

Doc vs Internet + Library

User ID: 78686

File name: ПЗ (Check copyright)

File ID: 1000663736

Page count: 76

Word count: 9966

Character count: 75245

File size: 5.55 MB

10% Matches

Highest match: 5.77% with library source. File ID:5972558

0.93% Internet Matches

19

Page 78

10% Library matches

129

Page 78

0% Quotes

No quotes found

0% Exclusions

No exclusions found

Replacement

No replaced characters found