

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**

**Кафедра** \_\_\_\_\_ **Комп'ютерних систем та мереж** \_\_\_\_\_

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач кафедри  
комп'ютерних систем та мереж

\_\_\_\_\_ Жуков І.А.

«\_\_\_\_\_» \_\_\_\_\_ 2021 р.

**ДИПЛОМНИЙ ПРОЕКТ**  
**(ПОЯСНЮВАЛЬНА ЗАПИСКА)**

**ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ**  
**“БАКАЛАВР”**

**Тема:** \_\_\_\_\_ **Комп'ютерна система обліку тренувань в тренажерному залі** \_\_\_\_\_

**Виконавець:** \_\_\_\_\_ **Бабенко Д.С.** \_\_\_\_\_

**Керівник:** \_\_\_\_\_ **Дрововозов В.І.** \_\_\_\_\_

**Нормоконтролер:** \_\_\_\_\_ **Журавель С.В.** \_\_\_\_\_

**Київ 2021**

# НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютерних систем та мереж

Напрямок (спеціальність) 123 "Комп'ютерна інженерія"

(шифр, найменування)

ЗАТВЕРДЖУЮ

Завідувач кафедри

комп'ютерних систем та мереж

Жуков І.А.

«\_\_\_\_» \_\_\_\_\_ 2021 р.

## ЗАВДАННЯ на виконання дипломного проекту

Бабенка Дмитра Сергійовича

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи): Комп'ютерна система обліку тренувань  
в тренажерному залі

затверджена наказом ректора від "26" квітня 2021 року № 648 /ст.

2. Термін виконання проекту (роботи): з 17.05.2021 до 20.06.2021

3. Вихідні дані до роботи: 1) вимоги до віртуальної комп'ютерної мережі;  
2) необхідні до реалізації типи використання віртуальної комп'ютерної мережі.

4. Зміст пояснювальної записки (перелік питань, що підлягають розробці):

- 1) аналіз роботи тренажерних залів;
- 2) проектування бази даних системи;
- 3) описання розробки онлайн системи для автоматизації роботи адміністратора тренажерного залу.

5. Перелік обов'язкового графічного матеріалу:

Презентація Power Point

## 6. Календарний план

№ п/п	Етапи виконання дипломного проекту	Термін виконання етапів	Примітка
1	Провести аналіз літератури за темою дипломного проекту та аналіз існуючих систем	17.05.21 - 18.05.21	
2	Провести аналіз предметної області	19.05.21- 20.05.21	
3	Розробити структуру бази даних системи	21.05.21- 25.05.21	
4	Розробити програмні засоби для автоматизації роботи адміністратора тренажерного залу	26.05.21- 30.05.21	
5	Провести налаштування програмних засобів на сервері	31.05.21- 02.06.21	
6	Написати пояснювальну записку	03.06.21- 10.06.21	
7	Підготувати презентацію та графічні матеріали	11.06.21- 12.06.21	
8	Захистити дипломний проект	14.06.21- 20.06.21	

7. Дата видачі завдання «17» травня 2021 р. \_\_\_\_\_

Керівник дипломного проекту \_\_\_\_\_ Дрововозов В.І.  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_ Бабенко Д.С.  
(підпис студента)

## РЕФЕРАТ

Пояснювальна записка до дипломного проекту “Комп’ютерна система обліку тренувань в тренажерному залі”: 60 с., 37 рис., 21 літературне джерело, 1 додаток.

*PHP, MySQL, АВТОМАТИЗОВАНЕ РОБОЧЕ МІСЦЕ, АДМІНІСТРАТОР, БАЗА ДАНИХ, МОДЕЛЮВАННЯ ПРОЦЕСІВ, ПРОЕКТУВАННЯ СИСТЕМИ, ТРЕНАЖЕРНИЙ ЗАЛ*

**Об’єкт дослідження** – облік роботи тренажерного залу.

**Предмет дослідження** – автоматизоване робоче місце адміністратора тренажерного залу.

**Мета дипломного проекту** – реалізувати програмне забезпечення для реалізації автоматизованого робочого місця адміністратора тренажерного залу.

**Прогнози припущення щодо розвитку об’єкта дослідження** – використанні розробленого ПЗ у діяльності тренажерний залів.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ .....	6
ВСТУП .....	7
РОЗДІЛ 1 АНАЛІЗ РОБОТИ ТРЕНАЖЕРНИЙ ЗАЛІВ.....	9
1.1 Поняття і характеристика роботи CRM .....	9
1.2. Аналіз предметної області «Автоматизоване робоче місце адміністратора тренажерного залу» .....	17
Висновки до розділу.....	23
РОЗДІЛ 2 ПРОЕКТУВАННЯ БАЗИ ДАНИХ СИСТЕМИ .....	24
2.1. Інфологічне проектування .....	24
2.2 Логічне проектування.....	33
2.3. Нормалізація таблиць реляційної бази даних .....	37
Висновки до розділу.....	41
РОЗДІЛ 3 ОПИСАННЯ РОЗРОБКИ СИСТЕМИ ДЛЯ АВТОМАТИЗАЦІЇ РОБОТИ АДМІНІСТРАТОРА ТРЕНАЖЕРНОГО ЗАЛУ .....	43
3.1. Вибір мови програмування .....	46
3.2. Реалізація системи .....	47
3.3. Компоненти мобільного додатка .....	51
3.4. Внутрішня реалізація складання тренувань .....	55
Висновки до розділу.....	60
ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	64
ДОДАТОК А СХЕМА АЛГОРИТМУ ФОРМУВАННЯ ЗВІТУ .....	65

Кафедра КСМ				НАУ 21 03 46 000 ПЗ			
Виконав	Ббенко Д.С.			Комп'ютерна система обліку тренувань в тренажерному залі	Літера	Аркуш	Аркушів
Керівник	Дрововозов В.І.					5	60
Консульт.					123 КС 421Б		
Н. контроль	Журавель С.В.						
Зав. Каф.	Жуков І.А.						

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

<i>ARP</i>	–	<i>Address Resolution Protocol</i> (протокол канального рівня, служить для визначення MAC-адреси отримувача)
<i>BGP</i>	–	<i>Border Gateway Protocol</i> (основний протокол динамічної маршрутизації в Інтернеті)
<i>FDB</i>	–	<i>Forwarding DataBase</i> (таблиця комутації)
<i>MAC</i>	–	<i>Media Access Control</i> (унікальний ідентифікатор, який присвоюється кожному вузлу комп'ютерних мереж)
<i>OSPF</i>	–	<i>Open Shortest Path First</i> (протокол динамічної маршрутизації, який використовує Алгоритм Дейкстри)
<i>PAP</i>	–	<i>Password Authentication Protocol</i> (протокол простої перевірки автентифікації)
<i>PoE</i>	–	<i>Power over Ethernet</i> (технологія передачі харчування по <i>Ethernet</i> )
<i>RIP</i>	–	<i>Routing Information Protocol</i> (протокол маршрутної інформації)
<i>VLAN</i>	–	<i>Virtual Local Area Network</i> (віртуальна локальна мережа)
ВОЛЗ	–	волоконно-оптичні лінії зв'язку
ЄІП	–	єдиний інформаційний простір
КІС	–	корпоративна інформаційна система

## ВСТУП

На сьогоднішній день здоровий спосіб життя набуває все більшої популярності серед людей. Сучасна молодь частіше стала віддавати перевагу заняттям спортом, правильного збалансованого харчування, хорошему сну, прогулянок на свіжому повітрі, відмови від шкідливих звичок. Люди стали частіше додавати в своє життя фізичну активність - майже 80% росіян стверджують, що спорт так чи інакше присутнє в їх житті [1].

Однією з основних причин такої шаленої популярності здорового способу життя є «сидячий» вид діяльності. Життя саме по собі стала вимагати від більшості людей все менше зусиль: ми їдемо до роботи на автобусі / машині, сидимо в задушливих офісах, проводячи за комп'ютером велику частину дня, приходимо додому і знову ж беремо в руки ноутбук або сідаємо перед телевизором. Через це рано чи пізно можуть початися проблеми зі здоров'ям, а хворіти, природно, не хочеться нікому. Тому, люди стали «компенсувати» брак активності протягом дня походами в тренажерний зал, басейн, або заняттями в домашніх умовах.

Крім цього, похід в спорт зал в сучасному суспільстві стає модною тенденцією - хлопцям хочеться мати рельєфне мускулисте тіло, а дівчатам хочеться бути стрункими і підтягнутими.

Однак, необхідно усвідомлювати, що недостатньо мати одне лише бажання почати тренуватися. При підході до занять спортом основним завданням є складання плану тренувань: підбір вправ, розрахунок кількості підходів і повторень, планування прогресу в інтенсивності тренування і зміні ваги спорядження. Щоб самостійно розібратися у всьому цьому, новачкові буде потрібно досить багато сил і часу. Тому існує інший, більш розумний вихід: можна звернутися до фахівця - професійного тренера, який допоможе скласти індивідуальний тренувальний план, зорієнтує на перших порах в правильності виконання вправ, а також при необхідності підкоригує навантаження. Природно, це задоволення не безкоштовне, і за нього вам доведеться віддати чималу суму грошей.

Так що ж робити людині, у якого немає часу і коштів на те, щоб розібратися у всіх тонкощах складання персонального плану тренувань? Відповідь проста - необхідно скористатися мобільним додатком, яке зробить це за вас - підбере по вашим параметрам оптимальний комплекс вправ, підкаже скільки зробити повторень і підходів, продемонструє техніку виконання, а також допоможе стежити за прогресом.

Таким чином, метою даної випускної кваліфікаційної роботи є розробка мобільного додатка для складання індивідуальних програм тренувань з можливістю внесення даних про їх виконання.

Для досягнення поставленої мети були позначені наступні завдання:

- аналіз і опис предметної області, аналіз і порівняння аналогів;
- формулювання вимог;
- вивчення принципів розробки Android-додатків;
- реалізація програми.



## РОЗДІЛ 1

### АНАЛІЗ РОБОТИ ТРЕНАЖЕРНИЙ ЗАЛІВ

#### 1.1 Поняття і характеристика роботи CRM

Дійсно, а що таке CRM-система? Це питання ми чуємо протягом 11 років, від власників бізнесу, співробітників, розробників, швендь по сайту людей. І ось він нам був знову поставлено вже тут, на Хабре. Ми змінили звичкою відповідати на всі коментарі та проігнорували питаннят3k, Щоб відповісти на нього максимально розгорнуто. Виявилося не так-то просто. А ви знаєте, що таке CRM?

Єдиного канонічного визначення не існує - і це виходить з логіки самого типу програмного забезпечення - CRM-системи. Справа в тому, що спочатку CRM - це концепція управління взаємовідносинами з клієнтами, а вона у кожної компанії своя. Значить, і визначення CRM як софту у всіх теж своє, що ми як вендори по повній і отримуємо на вході. Ось неповний список того, як обзивають систему: «програма», «софтинка», «програма для продажів», «ERP», «діаграма Гантта», «планер», «контакт-менеджер». А між тим, практично будь-яка CRM поєднує в собі всі ці іменування (хіба крім ERP - це вже більш заморочена історія).

Тому почнемо з відповіді на питаннят3k- що таке CRM, щоб це пояснити програмісту.

Якщо говорити грубо, то це щось на зразок Jira для комерційної служби: менеджери з продажу і маркетологи ведуть в системі користувачів (клієнтів), заводять інциденти (угоди), прописують бізнес-процеси (як ми це робимо в рамках ITSM), потім будують звітність і рахують гроші.

Кафедра КСМ				НАУ 21 03 46 000 ПЗ			
Виконав	Ббенко Д.С.			Аналіз роботи тренажерний залів	Літера	Аркуш	Аркушів
Керівник	Дрововозов В.І.					9	60
Консульт.					123 КС 421Б		
Н. контроль	Журавель С.В.						
Зав. Каф.	Жуков І.А.						

А якщо міркувати професійно, то CRM-система - це центральне сховище інформації про клієнтів, угодах та інших впливах, у якого є призначений для користувача інтерфейс з набором функцій. Відповідно, CRM виконує саму звичну роботу: накопичує дані, зберігає дані, обмінюється ними, автоматизує процеси і логірує дії.

Якщо говорити з точки зору керівника компанії, то CRM-система - інструмент управління клієнтами і пов'язаної з ними інформацією, а також спосіб прискорити і впорядкувати всі бізнес-процеси. Зауважте, ні слова про контроль! Вся контролююча функція CRM - це звіти за підсумками місяця, аналітика і KPI (якщо вони впроваджені в компанії). CRM не стежить за співробітниками, не контролює трафік, що не палить піратські програми, які не січе бездіяльність працівника - для цього є ПО іншого типу.

Але більше всьогонамподобається визначення, яке ми вчитали у колег з Salesforce: CRM - це нервова система компанії.

Вона виникла не так давно - вже на початку 2000-х, коли ринок переповнився рішеннями всіх типів, які розробники вирішили назвати CRM. Насправді, все і є CRM, тільки вони різного типу і різних можливостей.

Головна класифікація, яку ви можете зустріти, це поділ усіх CRM-систем на три типи

операційні CRM- системи, в яких ведеться оперативна робота, автоматизуються процеси продажів і маркетингу, зберігаються угоди і документи, з якої проводяться розсилки і т.д. Звітність в таких системах - супутня, але не веде функціональність.

аналітичні CRM- системи, в яких превалює аналітика. Це можуть бути системи з гнучкими звітами і дизайнерами звітів, вони можуть включати модулі BI і OLAP. Як правило, це дорогі рішення, які підходять для складної інтеграції, наприклад, з білінгом стільникового оператора або платіжною системою інтернет-магазину. Буває, що аналітична CRM не містить частини функцій управління продажами, маркетингом і процесами, а служить для цілей сегментації, управління асортиментом і т.д.

Колабораційних (це калька), стратегічні CRM- системи, заточені під комунікації, спільну роботу, в тому числі з клієнтами і партнерами. У таких CRM-системах може передбачатися особистий кабінет клієнта, внутрішня соціальна мережа, корпоративний портал. В стратегічних CRM розвинений набір можливостей для управління проектами та завданнями.

Якщо мова йде просто про управління продажами (клієнти + угоди + аналітика), то це раніше називали SFA (sales force automation). Зараз важко знайти ПО, що відноситься до цього класу, а в Росії так просто неможливо. Та й ні до чого.

До сих пір є окремі CRM-щики, які заявляють, що вони операційні або аналітичні, але в цілому все три типу давно проникли один в одного і світом бізнес-софта правлять великі і маленькі універсальні рішення, в яких працюють всі: від ТЕЛЕМАРКЕТОЛОГ і продавників до техпідтримки і навіть розробників (у нас в RegionSoft, наприклад, немає ніяких багтрекерів і систем управління проектами - ми під це заточили нашу власну RegionSoft CRM).

За ступенем готовності до впровадження

Готові рішення- ви купуєте готові компоненти, ставите їх як є і адаптуєте бізнес-модель до встановленого ПО. Такий варіант щодо бюджетен і підходить невеликим компаніям зі стандартними процесами або компаніям, що бажають автоматизувати один-два аспекти роботи (наприклад, вести клієнтів угоди і здійснювати дзвінки). Хоча сьогодні в більшості рішень є можливості налаштування системи клієнтом і навіть інструменти для самостійної доопрацювання (наприклад, вRegionSoft CRMце дизайнер звітів, шаблони документів, конструктор бізнес-процесів, в деяких CRM - своя платформа розробки або API).

Готові рішення з доопрацюванням- ви купуєте готові компоненти і просите вендора доопрацювати систему під особливості вашого бізнесу у відповідність до технічного завдання. Насправді, це найбільш поширений варіант придбання, і кожен вендор готовий надати послуги з доопрацювання. Основне завдання клієнта - грамотно зібрати вимоги і викласти їх розробнику.

Повністю замовна розробка- довгий, тернистий і дорогий шлях впровадження CRM в компанії. Ви збираєте вимоги, описуєте необхідні процеси і функціональність, вендор (або компанія розробки на замовлення) створює детальне ТЗ і розробляє CRM-систему під вас. При цьому може виявитися, що ви даремно наполягли на такому розкладі, і цілком могли обійтися доопрацюванням базового рішення. При всій привабливості замовного ПО «саме під компанію» у нього є відчутний мінус - у разі зміни бізнесу або вимог цього кожна софтверний зміна буде також замовний розробкою - довгої і дорогої.

Збірні рішення (конструктори)- набирає популярність софт, існуючий в двох варіаціях: в першій ви вибираєте потрібні вам модулі і в будь-який момент можете докупити решту, у другій ви користуєтеся якимось базовим рішенням, а пошта, телефонія, аналітика, месенджери і т.д. купуються у третьої сторони і інтегруються з основним ПО (як правило, це віджети, плагіни і аддони з маркетплейсов постачальника базового рішення).

за ліцензуванню

SaaS- рішення, ліцензії яких ви орендуєте і платите за них щомісяця, як за інтернет або телефон. Це вигідна вендору і не дуже вигідна компанії схема: поки у вас є CRM, ви за неї платите - рік, два, п'ять ... Ціна володіння виходить просто космічної. Так, наприклад, якщо у вас всього 10 співробітників і платите ви за все 990 рублів за співробітника в місяць, ціна володіння за 5 років складе 594 000 рублей. Загалом, вважайте. Розрахунок, в основному, йде на те, що сучасні користувачі звикли до абонентської плати в усьому і вважатимуть такий варіант зручним. Хоча вам, звичайно, розкажуть про масштабування і гнучкість. До речі, у нас теж є оренда нашої CRM-системи, але вона не така кабальна - просто для тих, кому конче потрібна CRM-ка, але немає грошей на оплату відразу всієї покупки.

Софт on-premise- ліцензії на ПЗ ви придбаєте у власність і платите один раз. В цьому випадку ви застраховані від будь-яких негативних подій в бізнесі вендора. Надалі ви можете заплатити тільки за мажорні поновлення, якщо вирішите їх придбати.

Вільне ПЗ (опенсорс)- найцікавіша форма існування CRM-системи. Можна побути хардкорних ІТ-компанією, форкнуть відомий проект, запив свої фічі і брати гроші з інших за технічну підтримку (а у неї роботи буде по горло, гроші не маленькі). Можна купити готовий форк і платити за технічну підтримку, користуючись «безкоштовним ПО». Можна форкнуть, найняти програміста і зробити ідеальну CRM-систему під себе - це недешево і складно, але буває ефективно. Головне, завжди уважно дивіться, з якої ліцензії поширюється вільне ПЗ і за який стек вам доведеться платити програмісту.

За включеності вендора

вендорських рішення- CRM-систему продає і впроваджує сама компанія-розробник (російські системи майже всі такі). Це ідеальний варіант - ви нікому не переплачуєте і працюєте безпосередньо з тим, хто створив софт і готовий відповідати за нього, навчати, допрацьовувати, підтримувати і т.д.

партнерські рішення- CRM-систему продає системний інтегратор або сертифікований партнер. Актуально, в основному, для зарубіжних CRM-систем і продуктів 1С. Партнери можуть просто виступати посередником, а можуть допрацьовувати системи, робити складні впровадження, розробляти коннектори, плагіни і т.д. Гарні, якщо у вас складне впровадження, неможлива віддалена установка і доопрацювання, а в місті немає вендора. В цілому, це більше історія для великих компаній.

Рішення «під крилом сильної компанії»- ультра актуальні події CRM-ринку. Спершу свою CRM випустив Білайн (зі скандалом, невиразною функціональністю і дивним просуванням), на днях про CRM-системи повідомив Сбербанк (насправді, це гарне перероблене рішення одного з гравців російського ринку CRM). У порівнянні з Білайном, система Ощад цілком легкотравна, але просування на безкоштовності, звичайно ж, всього лише маркетинговий хід. Такий тип систем поганий по ряду причин: по-перше, це не профільний бізнес компанії, по-друге, це не CRM на благо бізнесу, а спосіб тримати своїх клієнтів при собі, давши їм ще один інструмент, який їх прив'яже до компанії, в По-третє, це потреба компаній в ваших даних (цілком можливо,

вони будуть аналізуватися і використовуватися для створення продуктів і послуг - поживемо-побачимо, це тільки наше припущення).

саморобки- прості CRM-ки, які створюються внутрішніми програмістами або фрілансерами. Як правило, це щось типу простого контакт-менеджера. Однак не варто недооцінювати такі рішення - іноді вони виходять вдалим. RegionSoft CRM теж колись була системою для внутрішнього користування, а за 11 років доросла до складної ERP в своїй максимальній редакції Enterprise Plus. Тут, в загальному, як пощастить.

За принципом розгортання

десктопні- CRM ставиться на сервер компанії (може бути просто сучасний ПК), а користувачам ставляться робочі додатки. Можливий термінальний віддалений доступ з будь-якого пристрою при наявності інтернет-з'єднання. Всупереч розхожій міфу, у десктопного софта немає ніяких обмежень для філіальної або холдингової структури компанії. Це найбезпечніший спосіб розгортання.

хмарні- користувачі працюють через веб-інтерфейс або з мобільного, а вся інформація зберігається в хмарі (по суті, на чужих фізичних серверах в віддаленому дата-центрі). Вважається, що це самий зручний варіант з точки зору адміністрування (не зовсім, але зараз не про це). Дійсно, вам не потрібно піклуватися про стан сервера, сховище даних і т.д. Однак за це ви платите крім усього іншого безпекою - дані в будь-якому хмарі можна скомпрометувати, було б бажання. До речі, купуючи хмарну CRM-систему, не забудьте уточнити, чи стягується додаткова плата за бекапи, як вони робляться, скільки зберігаються, який Аптайм у сервера. Швидше за все, ви відкриєте для себе нові фінансові нюанси.

Серверні з веб-інтерфейсом- всі дані зберігаються в вашому офісному хмарі (у вас на серверах), а користувачі заходять через звичний веб-інтерфейс. Це більш безпечне рішення ніж хмара, але і більш вразливе, ніж десктоп.

Зоопарк бізнес-софта

Ще одна проблема, яка не дозволяє точно визначити CRM як тип софта, це навколишній бізнес зоопарк програмних рішень. І деякі з них навмисно або

через незнання також називають CRM-системами. Давайте оглядово, буквально в пару рядків, стисло нагадаємо основні типи ПО, яке найчастіше зустрічається в бізнесі і перетинається з CRM. До речі, про ці та інших типах ми писали докладно.

Системи управління проектами- програмні рішення, які дозволяють управляти проектами, тобто планувати зайнятість членів команди і ресурси, розставляти пріоритети, визначати черговість робіт, ставити дедлайни. Як правило, в центрі таких систем не варто клієнт, а варто проект - внутрішній чи зовнішній.

Системи управління завданнями- лайт-версія попереднього типу. По суті, колективний планувальник з завданнями, термінами, статусами, відповідальними і нагадуваннями. Знову ж клієнт в таких системах виступає всього лише як атрибут завдання.

Офісні пакети- звичні нам текстовий редактор, табличний редактор і редактор презентацій. Не має ніякого відношення до CRM, але часто виступає в ролі сховища клієнтської бази (холівар на цю тему вже був).

Бухгалтерські та фінансові програми і облікові системи- вічний конкурент CRM-систем, їх же головний біль. З одного боку, в центрі облікових систем стоять фінанси і грошові потоки, з іншого, їх намагаються правдами і неправдами пристосувати під ведення клієнтів. З третього - з цим усім треба інтегруватися, якщо бізнес пред'являє такі вимоги до CRM. Основна проблема в тому, що у сисадмінів, продавників і бухгалтерів абсолютно різні сутності та об'єкти, з якими вони працюють. Тому взаєморозуміння досягти вдається рідко.

ERP- величезні складні системи планування ресурсів підприємства. Використовуються в компаніях з численними фінансовими потоками, в виробничих корпораціях, на заводах і т.д. Від CRM відрізняються тим, що ставлять в центр не клієнта, а ресурси і виробничі процеси компанії. Однак ніщо не заважає об'єднати можливості CRM і ERP в одній системі.

CMS- система управління контентом сайту і / або блогу. Не має ніякого відношення до CRM, плутають за співзвучністю.

ВРМ-системи- системи проектування і моделювання бізнес-процесів компанії. Дозволяють автоматизувати буквально все: від дрібної рутини до великих виробничих процесів. У чистому вигляді потрібні тільки великим спеціалізованим компаніям. У центрі таких систем - внутрішні бізнес-процеси компанії, їх налагодження та бездоганне функціонування. Як модуль CRM - дуже цікава і потрібна штука навіть для невеликих фірм. Обов'язково налаштуйте бізнес-процеси, корисно їх переглянути і автоматизувати.

Віртуальні АТС і комплексний VoIP- той тип ПО, про який говорять «нам звонилки вистачить». Дійсно, багато операторів IP-телефонії мають власні контакт-менеджери і навіть мінімальні CRM. Як правило, функціональності починає швидко не вистачати, тому краще брати нормальну розвинену CRM і інтегрувати її з телефонією і віртуальної АТС. Виходить добре. А головне, затребуване багатьма клієнтами - настільки, що, наприклад, ми розробили і власний SIP-фон, і свій VoIP-коннектор для маршрутизації телефонії.

Білінгові системи- системи, в яких враховуються і аналізуються платежі клієнтів (не плутати з процесинговими платіжними системами). При необхідності інтегруються з CRM-системою.

аналітичні системи- зустрічаються, в основному, у великих компаніях. Це системи обробки надходять від клієнтів даних. можуть бути вендорських, а можуть самописні. Найпростіший варіант - всі дані накопичуються в СУБД, і програмісти готують замовні вивантаження для співробітників: пишеться SQL-запит, вивантажується масив, передається зацікавленій менеджеру. Найскладніший - дорогі імпорتنі рішення типу SAP.

ЕСМ-системи (вони ж СЕД)- системи електронного документообігу з формуванням, зберіганням, узгодженням і архівуванням документів. Зазвичай використовуються самі по собі, вкрай рідко бувають інтегровані з CRM або входять до її складу.

Корпоративні портали і корпоративні соціальні мережі- раніше були націлені виключно на спілкування співробітників всередині компанії і ведення корпоративної бази знань, сьогодні обросли можливостями CRM і впритул до них наблизилися. Великий мінус - безліч відволікаючих моментів під час



роботи в інтерфейсі і обмеженість CRM-ного функціоналу. Плюс - все в одному, співробітникам не потрібно перемикатися між програмами.

Так що, впроваджуючи корпоративне ПО, головне, не переплутати і вибрати потрібні рішення виходячи з розмірів та потреб бізнесу.

Так що з визначенням? Ми не претендуємо стати істиною в останній інстанції, тому просто запропонуємо свій варіант. CRM-система - це програмне забезпечення, що служить інструментом накопичення, зберігання і розвитку клієнтської бази, а також проектування і автоматизації бізнес-процесів в компанії. Хоча, повторимося, для кожної компанії мети впровадження свої і в підсумку визначення теж своє. Тому дамо неформальне, але глобальне визначення: CRM - це потрібна бізнесу програма, яка прискорить процеси, допоможе обійти конкурентів і збереже ваш найцінніший актив - клієнтську базу. Звичайно, якщо ви самі цього захочете.

## **1.2. Аналіз предметної області «Автоматизоване робоче місце адміністратора тренажерного залу»**

Все, що можна автоматизувати, має бути автоматизовано. Ця проста істина, але кожен, хто її засвоїв, вже виграв. Правильно підібрана програма для фітнес клубу вивільняє час для більш важливих завдань, а значить дозволяє заробити більше. Тому сміливо беріть в роботу досягнення ІТ-сфери, і ваш фітнес клуб буде нарощувати обороти.

Простий тест на необхідність введення автоматизації:

- ви керуєте фітнес клубом;
- в вашому підпорядкуванні понад 7 осіб.

Якщо на обидва питання ви закивали «так», то вам вже необхідна програма з управління фітнес клубом. У класичному менеджменті вважається, що управління колективом до 7-8 чоловік можна утримувати «в пам'яті», з іншого боку, програми управління допомагають організувати клієнтів.

Все ж наша позиція така: для невеликого фітнес клубу з колективом менше 7 осіб програма автоматизації бажана, але не обов'язкова. В інших випадках вона необхідна.

За допомогою комплексної автоматизації можна ефективно управляти фітнес центром, прогнозувати приплив або відплив клієнтів, відслідковувати попит і формувати пропозицію.

Отже, що вам дасть автоматизація фітнес клубу:

- облік клієнтів, відвідувань і ідентифікацію карт;
- контроль за оплатою послуг через клубну карту;
- можливість обліку послуг, що надаються за кількістю відвідувань або на певний період;
- управління тарифами на різні послуги клубу;
- фінансовий аналіз;
- планування графіка роботи співробітників;
- управління запасами на складі;
- формування звітів про діяльність клубу.

Система управління фітнес клубу

Програм дуже багато. У кожної є свої особливості і переваги (куди без недоліків, і це дуже суб'єктивна оцінка).

Щоб ви розуміли, з чого вибирати, ми зібрали всі найзручніші, популярні та зрозумілі системи управління фітнес клубом.

1С: Фітнес клуб: можливо, сама універсальна і звична з програм

Система 1с: Фітнес-клуб використовується для ефективного управління спортивними об'єктами будь-яких розмірів: від маленьких фітнес-студій до федеральних і міжнародних мереж. Вона дозволяє збирати детальну інформацію про членів клубу, використовувати її для збільшення продажів і продовжень, управляти членством і пакетами послуг і багато іншого завдяки вбудованому модулю галузевої CRM-системи.

1С: Фітнес клуб - зручний інструмент для автоматизації роботи всіх співробітників: від керівника до тренера, завдяки додатковим мобільним додатків. Інтеграція CRM-системи з особистим кабінетом клієнта, онлайн-розкладом і електронними картами Wallet дозволяє підвищити середній чек і збільшити кількість записів на заняття. А аналітичний сервіс фітнес-індустрії допомагає бачити відвідуваність, середню вартість абонементу, кількість

проданих абонементів та інші дані фітнес-об'єктів в вашому регіоні в реальному часі і приймати важливі рішення на основі реальних об'єктивних показників галузі.

1С: Фітнес-клуб дозволяє вести облік фінансових надходжень і переміщень коштів в касі і на банківському рахунку, працювати з депозитами і бонусними рахунками клієнтів. Також є можливість підключити інтеграцію з онлайн-касою.

Вести повну бухгалтерію фітнес-клубу можна завдяки прямій безкоштовної інтеграції з 1С: Бухгалтерія 8.

Демонстрація функціоналу «1С: Фітнес клуб» на офіційному сайті: [www.fitness1c.ru](http://www.fitness1c.ru)

### Програма для фітнес клубу PerfectGym

Зверніть увагу на цю систему! Цей пакет програмного забезпечення вважається одним з кращих для фітнес центрів, оскільки розроблений спеціально для цього типу бізнесу. Ця система управління відкриває доступ до розширеної аналітики та звітності, за допомогою якої можна збільшити відсоток утримання клієнтів, прогнозувати їх відтік і підвищити рівень задоволеності членів клубу.

PerfectGym володіє зручними інструментами для збору даних про клієнтів і їх обробки. Ще ви зможете контролювати фінансовий потік і створювати бухгалтерську звітність.

Якщо ви давно в гаджетах і рідко добираєтеся до комп'ютера, вам буде приємно дізнатися, що у PerfectGym є мобільний додаток. Використовуючи його, ваші клієнти зможуть самостійно записуватися на групові заняття, проводити платежі і управляти своєю клубною картою по її номеру. Неймовірно зручно, спробуйте.

### Хмарний сервіс FitBase: автоматизація фітнесу-клубу "під ключ"

FitBase - це новий сервіс для автоматизації та управління фітнес-клубом з вбудованою CRM.

«Рішення все в одному» - головна перевага хмарного сервісу Fitbase. Власники зможуть відстежити фінансові показники клубу в будь-який момент

часу, керуючі контролювати діяльність персоналу, маркетологи отримують нові рекламні інструменти, менеджери з продажу точно знати кому потрібно зателефонувати, а тренери управляти своїм розкладом тренувань.

Щоб виключити людський фактор в Фітбейс розроблений унікальний конструктор автоматичних дій, завдяки якому можна налаштувати створення завдань або SMS-повідомлень при виконанні різних тригерів: день народження клієнта, що наближається дата завершення абонементу, довга відсутність клієнта в клубі і так далі.

За допомогою віджетів особистого кабінету і онлайн-запису клієнти клубу зможуть оплачувати послуги з сайту і записуватися на персональні і групові заняття самостійно, скорочуючи навантаження на адміністративний персонал фітнес-клубу.

У Фітбейс є інтеграції з сайтом, SMS-шлюзами, IP-телефонією, СКУД (Pocketkey і Sigur), касовим та банківським устаткуванням.

Хмарний сервіс має простий і зрозумілий інтерфейс, установка займає не більше 15 хвилин, а для роботи не потрібно встановлювати сервер в клубі і наймати системного адміністратора. Є можливість доопрацювання додаткового функціоналу за бажанням клієнта.

FitBase підходить для автоматизації як великий мережі фітнес-клубів, так і для маленьких фітнес-студій, бійцівських клубів, спортивних шкіл або басейну.

Детальніше читайте на офіційному сайті <https://fitbase.io/>

Вирішуємо всі завдання фітнес-клубу з UNIVERSE Фітнес

Основне завдання галузевого продукту полягає в тому, щоб надати керівникам повну картину діяльності їхнього бізнесу. Чи не буде сфери, яка залишиться поза увагою UNIVERSE Фітнес.

Отже, які питання можна вирішити за допомогою UNIVERSE Фітнес:

- автоматизація роботи рецепції;
- ведення клієнтської бази;
- ведення фінансових розрахунків;
- складання оптимальних графіків роботи;

- запуск індивідуальних розсилок з урахуванням переваг клієнтів;
- контроль фінансових потоків;
- контроль ефективності роботи співробітників.

UNIVERSE Фітнес може синхронізуватися з іншим програмним забезпеченням і сумісно з різним обладнанням.

Бути всередині клубу разом з «КРАФТ: Фітнес-клуб»

Система управління фітнес клубу «КРАФТ: Фітнес-клуб» дозволяє автоматизувати роботу спортивних і SPA-центрів, а також інших закладів на клубній основі. Це програмне забезпечення адаптується до конкретних умов діяльності клубу. Воно дозволяє вести облік даних про клієнтів, складати прейскурант послуг та графіки роботи співробітників, а також відстежувати касові операції.

Для автоматизації роботи рецепції рекомендується використовувати цю програму в комплексі з модулем «КРАФТ: Торгівля». З його допомогою ви зможете вести повноцінний складський облік і бухгалтерію.

Просто і швидко управляти фітнес клубом разом з LuckyFit

Програмне забезпечення LuckyFit максимально простий і зручний у використанні. З його допомогою можна побачити повноцінну картину діяльності вашого фітнес-центру, оцінити поточну ситуацію і розробити стратегію розвитку.

Можливості LuckyFit:

- створення абонементів і управління ними;
- обробка клієнтської бази;
- індивідуальні розсилки клієнтам;
- складання розкладу групових та індивідуальних занять;
- складання графіка роботи співробітників;
- контроль доступу за рахунок інтеграції з турнікетами і електронними замками;
- реєстрація відвідувань;
- товарний облік;

- розрахунок зарплати тренерів;
- бізнес-аналітика.

Цей програмний продукт дозволяє керівникам віддалено стежити за діяльністю клубу через хмарну систему. Тому навіть у відпустці у вас не буде шансу відпочити від роботи, ні-ні, та й загляньте в систему. А за допомогою мобільного додатку LuckyFit ваші клієнти зможуть самостійно управляти своїми сервісами.

**Mobifitness: функціонально для власника, зручно для клієнта**

Mobifitness - це поширена система управління фітнес клубом, в якій об'єднані 4 важливі функції: облікова система, мобільний додаток для клієнтів, онлайн-запис для сайту і онлайн-запис для соціальних мереж. Цей програмний продукт дозволяє оптимізувати роботу співробітників, відстежувати статистику діяльності клубу і прогнозувати зміни кількості клієнтів.

Читайте кейс:

**Фітнес-клуб Athletic-GYM в мобільному телефоні**

Програма Mobifitness допомагає обробляти клієнтську базу і формувати пропозицію, виходячи з попиту на різні послуги фітнес-клубу. З її допомогою також можна автоматизувати роботу рецепції, складати графік роботи тренерського складу і розклад тренувань, вести облік фінансових операцій. Це програмне забезпечення інтегрується з різними операційними системами і пристроями контролю доступу.

**Fitness Pro: ідеально, якщо у вас мережа фітнес клубів**

Програма Fitness Pro може використовуватися як в одному клубі, так і в мережі фітнес центрів. З її допомогою можна вести клієнтську базу, продавати абонементи, контролювати баланс на них, проводити запис на групові заняття. При цьому в кожному фітнес клубі мережі можна буде перевірити дію абонементу.

**Функції Fitness Pro:**

- обробка даних і ведення клієнтської бази;
- відстеження відвідування занять і попит на окремі послуги клубу;

- складання графіка роботи персоналу і розкладу групових занять;
- товарний і складський облік, складання накладних;
- відстеження фінансових операцій;
- розрахунок заробітної плати за різними схемами;
- ведення звітності за певний період або за конкретної послуги;
- індивідуальний розсилки.

Автоматизація фітнес клубу - це вигідне рішення для вашого бізнесу. Ідіть в ногу з часом, полегшуйте роботу собі і своїм співробітникам, збирайте статистику і аналізуйте. Точніше, це зроблять за вас системи автоматизації фітнес-клубів. Навіть якщо вас бентежить, що частина функціоналу або повністю програма будуть платними, сміливо розлучайтеся з накопиченим і замовляйте програму для свого фітнес-клубу. Витрати окупляться дуже швидко, а ось користь ви отримаєте величезну.

### **Висновки до розділу**

Впроваджуючи CRM і мотивуючи співробітників її використовувати, керівник закриває безліч проблем і може бути впевнений, що його бізнес не розіб'ється об банальне небажання співробітників працювати на компанію.

CRM-система сьогодні - це наріжний камінь автоматизації малого і середнього бізнесу. Вона забезпечує організацію аналітикою, робить продажі інтелектуальними, скорочує час на рутинні завдання. Якщо є CRM, а в ній зібрані актуальні і докладні дані, то менеджери не вгадують, а аналізують, а значить завжди роблять твердий крок вперед.

## РОЗДІЛ 2

### ПРОЕКТУВАННЯ БАЗИ ДАНИХ СИСТЕМИ

#### 2.1. Інфологічне проектування

Якщо запитати айтишників, для чого в компанії впроваджується CRM-система, то чимала частка співрозмовників відповість, що, в основному, для мук і тортур системного адміністратора. Хтось ще розповість, що для знущань над будь-яким штатним айтишником або хоча б нещасним просунутим гиком в компанії. Це, звичайно, все жарти. А якщо серйозно, то часом навіть самі компанії, що вибрали CRM, не можуть відповісти на це питання. З популярного: «програма для продажів» і «система, яка продає». Питання «Для кого впроваджується CRM?» викликає ще більший розкид думок, серед яких лідирують «для боса», «для продавників» і, вважаємо, десь в списку «для галочки» і «для понтів». Тим часом CRM-система адресована цілком конкретним співробітникам, які можуть з її допомогою явити чудеса професіоналізму та ефективності - хоч в офісі, хоч на удалёнке. Якщо вони цього, звичайно захочуть. Так хто ці люди, заради яких намагаються сисадміни, програмісти і самі розробники CRM?

CRM в розумінні більшості - це програма для продажів. І дуже багато хто дивується, коли дізнаються, скільки можливостей і функцій впроваджено в сучасні CRM-системи: Тут і бізнес-процеси, і ABC-аналіз, і асортиментні матриці, і телефонія, і планувальники, і все що завгодно. Тому немає нічого дивного в тому, що в малому бізнесі в CRM може легко працювати вся компанія, причому кожен співробітник виконує свої завдання.

Кафедра КСМ				НАУ 21 03 46 000 ПЗ			
Виконав	Ббенко Д.С.			Проектування бази даних системи	Літера	Аркуш	Аркушів
Керівник	Дрововозов В.І.					24	60
Консульт.					123 КС 421Б		
Н. контроль	Журавель С.В.						
Зав. Каф.	Жуков І.А.						



Але тим не менше завжди є активні користувачі і є ті, хто звертається до системи від разу до разу. Так для кого в компанії впроваджується і працює CRM?

Взагалі в цьому місці ми в кращих рекламних традиціях повинні були використовувати кліше і пафосно відповісти, що CRM в компанії виключно для клієнта. Але за цими словами не варто нічого, крім маркетингового виверту, тому що CRM в компанії - це інструмент співробітників, за допомогою якого вони працюють з клієнтом і отримують виручку для компанії. Насправді, можна мати SAP або Salesforce з омніканальністю і при цьому втрачати клієнтів і ловити негатив, а можна і без CRM бути коханою компанією клієнтів (на жаль, уже майже не можна - на тлі інформаційного шуму про вашу крутизні без CRM ви просто не впораєтеся з потоком клієнтів і відразу станете поганими). CRM - це класний і зручний інструмент для швидкого виконання завдань, скорочення рутини, налагодження процесів і успішної оперативної роботи.

Для відділу продажів

CRM-система- це не просто права рука відділу продажів, це буквально відділ цілком, оскільки абсолютно вся діяльність менеджерів з продажу зав'язана на CRM. У неї надходить перша інформація про Ліді, з її допомогою здійснюються дзвінки, проводиться інформування клієнта, ведеться листування, в ній же угода проходить всі стадії і успішно закривається або, навпаки, йде в незавершені і тоді у CRM з'являється нове завдання - нагадати про незавершеною угоді, щоб менеджер міг повернутися до обговорення питання пізніше. Загалом, все що продажник звик тримати в голові або никатися по замітках і табличок, є в CRM, ну а голову можна звільнити для пошуку адекватної стратегії продажів і відбудови від загальносвітових традицій впарівання, нав'язування і дожима. (До речі, якщо ви думаєте, що в ІТ немає місця явищам практично агресивних продажів, то ви помиляєтеся: деякі компанії після уваги до них знайдуть вас в соціальних мережах, в інстаграмме, в ютубі і де завгодно. З одного боку, маркетинг і ретаргетінг у них хороший, з

іншого - пора навчитися реагувати на прохання «не дзвоніть мені взагалі, я давно все купив у ваших конкурентів, встановив і радію»).

Отже, чим корисна CRM відділу продажів?

- Головне достоїнство CRM-системи - централізоване і пов'язане зберігання всієї інформації про клієнтів в єдиній базі. Менеджерам не потрібно збирати інформацію по всіляких записок, нотаток, таблицями та електронною поштою - все доступно рівно в один клік (базова інформація про клієнтів та угодах доступна в будь-який, навіть самої молодій і слабкій CRM-ке, в серйозних універсальних CRM-системах типу RegionSoft CRM рівно так само при натисканні доступні листування з клієнтом, записи розмов з ним, вся сформована за нього документація, складські залишки, реєстр доставки кореспонденції та ін). Інформація уніфікованість, легко збирається у вигляді звітів на основі готових форм і шаблонів, тому поточна робота з нею скорочується часом до декількох секунд: відкрив шаблон документа - вибрав мишкою параметри і дані - перевірів - натиснув «Готово» і вивів на друк або відправив по емейл .

- Не менш важлива перевага - наявність звітів з продажу, по клієнтах, товарам і послугам, воронка продажів і т.д. Завдяки цим звітам менеджери з продажу можуть аналізувати профілі, продуктовий асортимент, вибудовувати стратегію, здійснювати допродажу і т.д. Фактично звіти - це кров всієї системи продажів, тому що всі ідеї і знахідки для збільшення виручки як правило криються саме в аналітиці, а не в інтуїції керівника відділу продажів (якої у нього часто і немає).

- процес продажів - це не те що ви бачили у фільмі «Вовк з Уолл Стріт»: по-перше, тому що у вас інша сфера діяльності і інший час, а не біржа кінця 80-х, по-друге, тому що такий стиль продажів мало кому підходить (особливо в IT!), по-третє, це всього лише фільм. Так ось, процес продажів - це рутинний процес з алгоритмом в підставі і опціями у вигляді досвіду або харизми продавця. Але алгоритм повинен бути універсальним, щоб продавали все і стабільно, а компанія б не жила від однієї зоряної угоди до іншої. CRM допомагає перетворити продажу в бізнес-процес: чіткий, налагоджений, з

етапами і тригерами (лід прийшов - лід кваліфікували - перший дзвінок - результат - фоллоуап-лист - новий дзвінок - презентація - угода (з купою своїх компонентів)). Це значно впорядковує роботу. Існування покрокового алгоритму продажів, вбудованого в CRM,

- А тепер уявіть ситуацію. У вас в компанії три продавця, у кожного своя тактика, повна зайнятість і загальна таблиця Excel з даними про продажі (в якій сам чорт ногу зломить, другу вивихне). Ви наймаєте четвертого співробітника і його потрібно ввести в курс справи. Всім колись, техніки у всіх різні, в даних зорієнтуватися він не може, в результаті людина простоює і втрачає лояльність до роботодавця через високого рівня стресу. А CRM - відмінний інструмент для адаптації новачків. У перший же день йому можна дати права доступу до бази знань, огляду бізнес-процесів і окремих операцій і надати можливість почати навчання, а заодно познайомитися з інтерфейсом CRM-системи, в якій йому належить працювати.

До слова, саме в кризовий період, такий як зараз, CRM-система знаходить додаткову цінність, так як вона безпечно зберігає всю комерційну інформацію, дозволяє ефективно працювати віддаленої і розподіленої команді і в той же час надійно логірує всі дії (не всі логірують, тут, зокрема мова йде про RegionSoft CRM). Якщо хтось із співробітників під прикриттям домашньої даху вирішить підзаробити на допомогу конкурентам або ретельно підготуватися до зміни роботи при збереженні клієнтської бази, то повз увагу CRM ці факти не пройдуть і ви збережете цінний ресурс - актуальну клієнтську базу.

#### Для маркетингу і PR

Коли ваш маркетолог в черговий раз попадеться вам на очі або вийде на зв'язок в ZOOM, запитайте у нього, хто ваша цільова аудиторія і як вона зараз себе почуває в зв'язку з ковідо-економіко-політико-соціокультурним кризою. Якщо він дасть чіткий розклад, любите і бережіть його, якщо немає, нехай сідає за аналітику і шукає відповідь на питання.

Жодна стратегія маркетингу (навіть якщо це ось такий контент, блог на Хабре і т.д.) не будується без аналітики - навіть (тим більше!) В категоріях штучного товару, люкса і предметів розкоші. CRM- золота інформаційна

жиладля хорошого маркетолога і піарника. Маючи доступ до профілю продажів, до записів про переговори з клієнтом, до питань і претензій клієнтів, до даних про цикл продажів і періоді до прийняття рішення, маркетолог може творити чудеса і готувати вдалі цільові стратегії, а не фантазувати.

Орієнтація була 100% на чоловіків - логічно ж. До тих пір, поки одна клієнтка не спитала, чи можуть вони упакувати аж три замовлення і відправити за трьома адресами: двом братам і татові. Блискавкою в голову прийшла гіпотеза, провели дослідження. З'ясувалося очевидне: дівчата замовляють приналежності своїм чоловікам набагато частіше, ніж ті роблять це самі (70/30). Поміняли розсилки, змінили рекламу, продажі вирости. Магазин через три роки закрили, господар поїхав жити в теплі країни і робити новий бізнес на зож-напоях, інвестувавши прибуток від першого. Він перейшов в буддизм, але не змінив віру в CRM і ретельно стежить за даними.

CRM допомагає проводити розсилки клієнтам і робити допродажу або стимулювати закриття відкладених угод. Наприклад, в RegionSoft CRM (Як в більшості топових CRM-систем) можна створювати вибірки для розсилок: групи клієнтів по будь-якою ознакою отримують свої, персоніфіковані листи з релевантним пропозицією. Такий підхід гарантовано збільшує конверсію і допомагає заробити на допродаж.

CRM-системи вміють «підтягувати» дані з електронної пошти, чатів, веб-форм. Це вкрай зручно для збору маркетингової інформації різного роду. Правда, іноді вендори вбудовують в систему омніканальність, що значно здорожує систему, але не має особливого толку зокрема для сфери B2B, коли клієнти уникають звернення через соціальні мережі та численні месенджери. Тому завдання маркетолога не тільки взяти цей інструмент, але і не переборщити.

Хороший маркетолог скористається даними про продажі в CRM і зможе зробити висновки про задоволеність клієнтів, їх проблеми і складнощі. Виходячи з цього можна буде виокремити продуктове рішення, яке закрий ці питання.

Cloudswave підрахував, що використання CRM може знизити маркетингові витрати компанії на 23%, і очевидно, що цифра цілком досяжна. Тим часом, в компаніях, що впровадили CRM, нерідко йде боротьба маркетингу з продажами і важливо простежити, щоб у всіх був доступ до необхідних для роботи даними. CRM-система ні в якому разі не повинна узурпувати одним підрозділом і використовуватися як зброя внутрішньокорпоративної боротьби. Це загальні дані для спільної праці.

### Служба підтримки

Надання доступу в CRM-систему для співробітників підтримки в деяких компаніях - велика загроза безпеки, тому що співробітники саппорта змінюються частіше за інших і на вході в компанію перевіряються дещо менше. Таким чином, можуть відбуватися витік даних, пов'язані з людським фактором. Проте, якщо у вас «збита банда» техпідтримки і ви не міняєте підтримку, як оператори зв'язку, CRM може стати найкрутішим інструментом для обслуговування клієнтів - набагато крутіше хелпдесков. Причина все та ж: величезний пласт інформації.

Ось що може саппорт з CRM напереваги.

- CRM персоналізує обслуговування клієнтів. І справа не в тому, що менеджер на вхідному або вихідному дзвінку зможе звернутися до клієнта по імені - справа в тому, що знання історії попередніх контактів і звернень в продажі і в сервіс дозволить не витратити час на з'ясування подробиць, оскільки вся інформація буде перед очима. Фактично менеджери «бачать» людини, який прийшов зі зверненням. І якщо для нас з вами це проста технологія, то для клієнта - прояв поваги і магія.
- CRM значно скорочує час відгуку співробітника підтримки: всі дані про клієнта доступні по одному кліку і немає необхідності піднімати історію в якомусь іншому файлі або папці. Саппортёр ще на початку спілкування знає про придбаних клієнтом продуктах, історії взаємодії, питаннях і іноді навіть про профіль використання продукту компанії - завдяки цій інформації підтримка надається ефективно, швидко і за програмними цілями, без перекладів від фахівця до фахівця.

- Маючи CRM в компанії, можна вибудувати систему вір-менеджерів, які не роздуваючи штат - завдяки доступу до даних і збору актуальної інформації в системі по кожному клієнту, кожен співробітник підтримки працює як персональний менеджер, який знає про клієнта все. Ну а кожного клієнта приємно бути вір.

- В CRM зберігаються всі чати і переговори з клієнтом, а значить, вона може бути інструментом контролю якості та допомагати розібратися в спірних ситуаціях, виключивши або підтвердивши вину менеджера (виникла негативна клієнтська реакція - прослухали розмова - зробили висновки - проаналізували - працюємо далі).

Додаткова перевага роботи саппорта в CRM - це те, що вся інформація з цього каналу взаємодії буде знаходитися все в тій же єдиній інформаційній базі і інші підрозділи зможуть включити її в аналітику.

Нарешті, CRM для керівника

Немає нічого дивного в тому, що в нашій статті керівник виявився в кінці. CRM-система для нього найчастіше - великий детальний дашборда, масив цифр, які він може подивитися, сформував в звіт і інтерпретувати. Якщо керівник сам не продає, не готує розсилки і не є багаторуки Шивой, CRM для нього пункт спостереження за оперативною діяльністю компанії (не плутайте - не для спостереження за ходом роботи підлеглих!).

Якщо в CRM є KPI, Це велика знахідка серед бізнес-софта. Якщо KPI реалізований правильно і грамотно, вважайте, джек-пот. Я ніколи не писав, що наша CRM найкраща (по-перше, ФЗ «Про рекламу», по-друге, в чомусь хороші ми, в чомусь конкуренти), але зараз можу сміливо сказати: з російськомовних CRM система KPI в RegionSoft CRM реалізована просто відмінно. У RegionSoft CRM впроваджена унікальна потужна система KPI, ноу-хау, придумане і розроблене з нуля ... Особливістю цього механізму є можливість створення профілів на основі заздалегідь визначених показників, яких показників, розрахованих на основі скриптів. Така технологія дозволяє налаштувати KPI як самому користувачеві без програмування, так і програмісту з просунутими навичками. Як це реалізовано, простіше подивитися на відео:

Воно довге, але для більшості компаній може виявитися корисним, тому що КРІ - відомий камінь спотикання як в ІТ, так і в інших галузях

CRM - це інструмент безпеки. Ви можете регулювати рівень доступу співробітників до даних, відстежувати дії в CRM (формування документів, звітів, дії з таблицями і т.д.), можете встановити додаткові засоби захисту для придбаної системи.

CRM ефективно збирає і зберігає дані. Ваша найцінніша інформація, дані клієнтської бази, що не розподіляється по декількома ІТ-систем, а знаходиться в єдиному інформаційному масиві і тим самим полегшує зберігання, обробку і захист корпоративних даних. Спільно «зароблені» компанією дані спільно же і використовуються, а не живуть в поштах, мобільниках і головах окремих співробітників.

Нарешті, за допомогою CRM ви можете забезпечити економічну безпеку компанії: менеджери ведуть клієнтів в CRM-системі, а не приховують їх від керівництва і колег, вважаючи їх своєю власністю (заробленої на чужі кошти і оплаченої роботодавцем - але хто про це думає!).

Перелік сутностей бази даних в таблиці 2.1.

## Перелік сутностей предметної області

№	Назва та позначення сутності	Ключ сутності і його позначення	Атрибути сутності та її позначення
1	ВидиКарт	<i>ID</i>	Вид вартість термін ВидЗнижки знижка ЧасЗ ЧасПо Пн Вт Ср Чт Пт Сб Нд
2	Кarti	<i>N</i> Кarti	Вид <i>ID</i> активна видана Діє до
3	Клієнти	<i>ID</i>	Прізвище ім'я По батькові Дата народження телефони Адреса договір фото Кіл_ть Візитів Активний
4	Відвідування	Дата час Клієнт <i>ID</i>	Зал <i>ID</i> <i>N</i> Ключа Коментар
5	ПоперЗапис	Дата час Клієнт <i>ID</i>	Зал <i>ID</i> Коментар
6	Зали	<i>ID</i>	Зал
7	КartiКлієнта	Клієнт <i>ID</i> <i>N</i> Кarti	



## 2.2 Логічне проектування

Зв'язок «багато-до-багатьох» між сутностями «Карти» і «Клієнти» представлено на рис. 3.1.

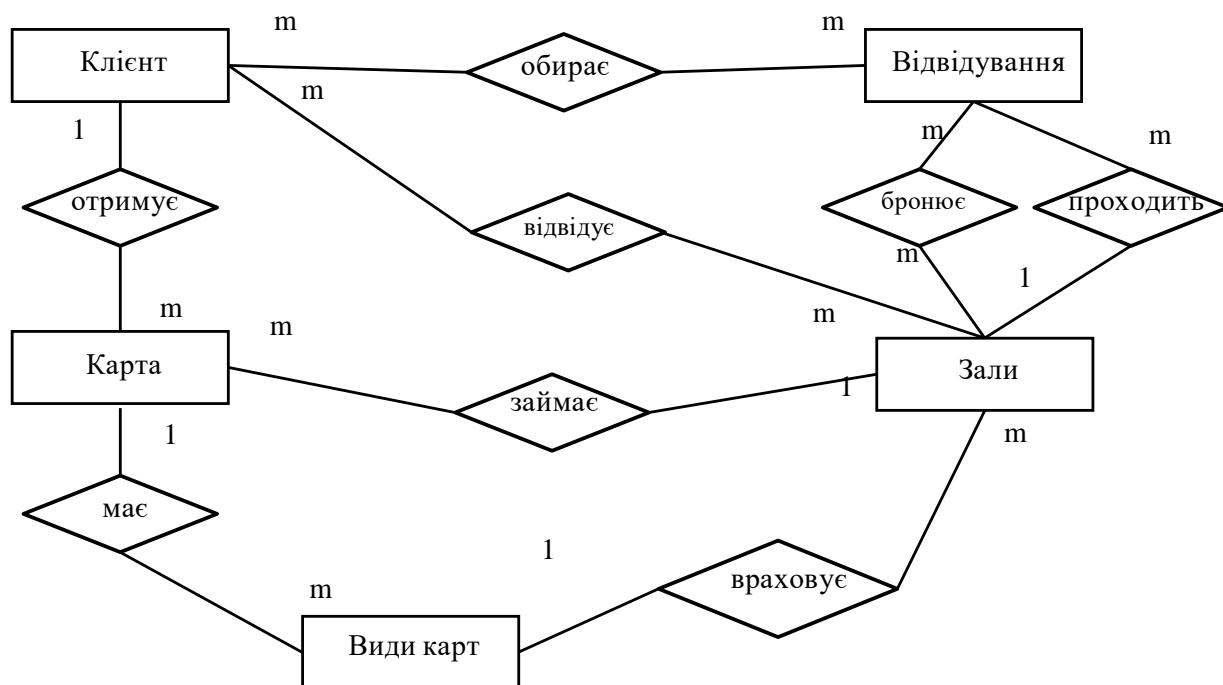


Рис. 2.1. ER-діаграма системи

Структура таблиць з детальним описанням типів полей наведена в табл. 2.1-2.7.

При описанні таблиць використано введено поле підстановки, яка описує тип зв'язку з зовнішніми даними. Так код:

```
SELECT [ВидиКарт]. [ID],  
[ВидиКарт].[Вид]  
FROM ВидиКарт;
```

Отримує список всіх видів карт, які представлені у таблиці ВидиКарт.

Структура таблиці «ВидиКарт»

Ім'я поля	Тип
<i>ID</i> (ключ)	Лічильник
Вид	текстовий
Вартість	грошовий
Термін	числовий
ВидЗнижки	логічний
Знижка	числовий
ЧасЗ	Дата час
ЧасПо	Дата час
Пн	логічний
Вт	логічний
Ср	логічний
Чт	логічний
Пт	логічний
Сб	логічний
Нд	логічний

Таблиця 2.2

Структура таблиці «Карти»

ім'я поля	Тип	підстановка
<i>N</i> Карт (ключ)	Лічильник	
Вид <i>ID</i>	Числовий	<i>SELECT</i> [ВидиКарт]. [ <i>ID</i> ], [ВидиКарт].[Вид] <i>FROM</i> ВидиКарт;
активна	Логічний	
видана	Логічний	
Діє до	Дата час	

Структура таблиці «КартиКлієнта»

ім'я поля	Тип	Підстановка
КлієнтID (ключ)	числовий	<i>SELECT</i> [Клієнти]. [ID], [Клієнти]. [Прізвище], [Клієнти]. [Ім'я], [Клієнти]. [ПоБатькові]  <i>FROM</i> Клієнти;
NКarti (ключ)	числовий	<i>SELECT</i> [Кarti]. [NКarti]  <i>FROM</i> Кarti;

Структура таблиці «Клієнти»

Ім'я поля	Тип
ID (ключ)	Лічильник
Прізвище	Текстовий
ім'я	Текстовий
По батькові	Текстовий
Дата народження	Дата час
телефони	Текстовий
Адреса	Текстовий
договір	Текстовий
фото	Поле об'єкта <i>OLE</i>
КілВізитів	Числовий
активний	Логічний

Таблиця 2.5

## Структура таблиці «Відвідини»

ім'я поля	Тип	Підстановка
Дата (ключ)	Дата час	
Час (ключ)	Дата час	
КлієнтID (ключ)	числовий	<i>SELECT</i> [Клієнти]. [ID], [Клієнти]. [Прізвище], [Клієнти].[Ім'я], [Клієнти].[По Батькові] <i>FROM</i> Клієнти;
ЗалID	числовий	<i>SELECT</i> [Зали]. [ID], [Зали]. [Зал] <i>FROM</i> Зали;
№Ключа	числовий	
коментар	текстовий	

Таблиця 2.6

## Структура таблиці «ПоперЗапис»

Ім'я поля	Тип	Підстановка
Дата (ключ)	Дата час	
Час (ключ)	Дата час	
КлієнтID (ключ)	числовий	<i>SELECT</i> [Клієнти]. [ID], [Клієнти]. [Прізвище], [Клієнти]. [Ім'я], [Клієнти].[По Батькові] <i>FROM</i> Клієнти;
ЗалID	числовий	<i>SELECT</i> [Зали]. [ID], [Зали]. [Зал] <i>FROM</i> Зали;
коментар	текстовий	

Таблиця 2.7

## Структура таблиці «Зали»

Ім'я поля	Тип
<i>ID</i> (ключ)	Лічильник
зал	Текстовий

Таким чином, отримуємо завершену модель даних.

### 2.3. Нормалізація таблиць реляційної бази даних

Для проектування схеми бази даних, потрібно згадати 7 формальних правил і саму концепцію нормалізації і денормалізації. Вони і лежать в основі всіх правил проектування.

Опишемо більш детально 7 формальних правил:

ставлення один до одного:

1.1) з обов'язковою зв'язком:

прикладом може виступати громадянин і його паспорт: у будь-якого громадянина повинен бути паспорт; паспорт один для кожного громадянина

Реалізувати цю зв'язок можна двома способами:

1.1.1) в одній сутності (таблиці):

Тут таблиця Citizen є сутність громадянина, а атрибут (поле) PassportData містить всі паспортні дані громадянина і не може бути порожнім (NOT NULL).

1.1.2) в двох різних сутності (таблицях):

Тут таблиця Citizen є сутність громадянина, а таблиця PassportData - сутність паспортних даних громадянина (самого паспорта). Сутність громадянина містить атрибут (поле) PassportID, який посилається на первинний ключ таблиці PassportData. У свою чергу сутність паспортних даних містить атрибут (поле) CitizenID, яке посилається на первинний ключ CitizenID таблиці Citizen. Поле PassportID таблиці Citizen не може бути порожнім (NOT NULL). Також тут важливо підтримувати цілісність поля CitizenID таблиці PassportData, щоб забезпечити зв'язок один до одного. Іншими словами, поле PassportID таблиці Citizen і поле CitizenID таблиці PassportData повинні

посилатися на одні й ті ж записи як якби це була одна сутність (таблиця), представлена в пункті 1.1.1.

1.2) з необов'язковою зв'язком:

прикладом може виступати людина, що має або не має паспорт конкретної країни. У першому випадку він буде громадянином даної країни, а в другому - немає.

Реалізувати цю зв'язок можна двома способами:

1.2.1) в одній сутності (таблиці):

Таблиця Person є сутність людини, а атрибут (поле) PassportData містить всі його паспортні дані і може бути порожнім (NULL).

1.2.2) в двох сутності (таблицях):

Таблиця Person є сутність людини, а таблиця PassportData - сутність паспортні дані людини (самого паспорта). Сутність людини містить атрибут (поле) PassportID, який посилається на первинний ключ таблиці PassportData. У свою чергу сутність паспортних даних містить атрибут (поле) PersonID, яке посилається на первинний ключ PersonID таблиці Person. Поле PassportID таблиці Person може бути порожнім (NULL). Тут також важливо підтримувати цілісність поля PersonID таблиці PassportData. Це потрібно, щоб забезпечити зв'язок один до одного. Поле PassportID таблиці Person і поле PersonID таблиці PassportData повинні посилатися на одні й ті ж записи як якби це була одна сутність (таблиця), показана в пункті 1.2.1. Або ж дані поля повинні бути невизначеними, тобто, містити NULL.

Ставлення один до багатьох:

2.1) з обов'язковою зв'язком:

Прикладом можуть виступати батько і його діти. У кожного з батьків є як мінімум одна дитина.

Реалізувати цю зв'язок можна двома способами:

2.1.1) в одній сутності (таблиці):

Таблиця Parent представляє сутність батька, а атрибут (поле) ChildList містить інформацію про дітей. Дане поле не може бути порожнім (NOT NULL).

Зазвичай типом поля ChildList виступають неповно структуровані дані (NoSQL) такі як XML, JSON і т.д.

#### 2.1.2) в двох сутності (таблицях):

Таблиця Parent представляє сутність батька, а таблиця Child - сутність дитини. У таблиці Child є поле ParentID, що посилається на первинний ключ ParentID таблиці Parent. Поле ParentID таблиці Child не може бути порожнім (NOT NULL).

#### 2.2) з необов'язковою зв'язком:

Прикладом може виступати людина, у якого можуть бути діти або їх може не бути.

Реалізувати цю зв'язок можна двома способами:

##### 2.2.1) в одній сутності (таблиці):

Таблиця Parent представляє сутність батька, а атрибут (поле) ChildList містить інформацію про дітей. Дане поле може бути порожнім (NULL). Зазвичай типом поля ChildList виступають неповно структуровані дані (NoSQL) такі як XML, JSON і т.д.

##### 2.2.2) в двох сутності (таблицях):

Таблиця Parent представляє сутність батька, а таблиця Child - сутність дитини. У таблиці Child є поле ParentID, що посилається на первинний ключ ParentID таблиці Parent. Поле ParentID таблиці Child може бути порожнім (NULL).

2.2.3) в одній сутності з посиланням на саму себе за умови, що у сутностей (таблиць) батька і дитини будуть однакові набори атрибутів (полів) без урахування посилання на батька:

Сутність (таблиця) Person містить атрибут (поле) ParentID, який посилається на первинний ключ PersonID цієї ж таблиці Person і може містити пусте значення (NULL).

Також дана реалізація є прикладом реалізації відносини «багато до одного» з необов'язковою зв'язком.

відношення багато до одного:

Цей зв'язок можна розглянути дзеркально до наведеної вище зв'язку один до багатьох. Іншими словами, відношення сутності «діти» до суті «батьки», де обов'язкова зв'язок буде за умови, що у дитини є хоча б один з батьків. Якщо ж беруть участь всі діти, в тому числі і що знаходяться в дитячих будинках, ставлення буде з необов'язковою зв'язком.

відносини багато до багатьох:

Прикладом може виступити нерухомість: вона може бути у власності як одну людину, так і декількох. З іншого боку, одна людина може володіти декількома будинками або частками кількох будинків.

Реалізувати дане відношення, із залученням NoSQL, можна так само, як в описаних вище відносинах. Однак, в рамках реляційної моделі зазвичай таке ставлення реалізують через 3 сутності (таблиці):

Таблиці Person і RealEstate представляють відповідно сутності людини і нерухомості. Зв'язуються дані сутності (таблиці) через сутність (таблиці) PersonRealEstate. Атрибути (поля) PersonID і RealEstateID посилаються на первинні ключі PersonID таблиці Person і RealEstateID таблиці RealEstate відповідно. Зверніть увагу, що для таблиці PersonRealEstate пара (PersonID; RealEstateID) завжди є унікальною і тому може виступати первинний ключем для самої сполучною сутності PersonRealEstate.

Також дане відношення можна реалізувати через більш ніж три сутності. Для цього додаються необхідні атрибути, які посилаються на первинні ключі необхідних відповідних сутностей. Така реалізація схожа з прикладами, описаними вище в пунктах 1.1.2 і 1.2.2.

Відносини один до багатьох і багато до одного можна реалізувати через більш ніж дві сутності. Для цього додаються необхідні атрибути, які посилаються на первинні ключі необхідних відповідних сутностей. Така реалізація схожа з прикладами, описаними вище в пунктах 1.1.2 і 1.2.2:

п.1 (п.1.1 та п.1.2) - перше і друге формальні правила

п.2 (п.2.1 та п.2.2) - третє і четверте формальні правила

п.3 (аналогічно п.2) - п'яте і шосте формальні правила



#### п.4 - сьомий формальне правило

У тексті вище ці сім формальних правил об'єднані в чотири блоки по функціоналу.

Говорячи про нормалізацію, потрібно розуміти її суть. Нормалізація веде до зменшення повторюваності зберігання інформації, а отже і до зменшення можливості появи аномалій у даних. Однак, нормалізація при дробленні сутностей призводить до більш складним побудов запитів для маніпуляцій з даними (вставки, модифікації, вибірки і видалення).

Зворотним процесом нормалізації називається денормалізація. Це спрощення побудови запитів доступу до даних за рахунок укрупнення і вкладеності сутностей (наприклад, як було показано вище в пунктах 2.1.1 і 2.2.1 за допомогою неповно-структурованих даних (NoSQL)).

#### **Висновки до розділу**

Основне завдання вбудованого CRM модуля для фітнесу - це надання адміністраторам і менеджерам відділу продажів фітнес клубу таких механізмів, за допомогою яких вони зможуть оперативно і своєчасно виконувати свої обов'язки. Якщо говорити про фітнес індустрії, то основні функції, які повинна виконувати CRM система - це ведення клієнтської бази з можливістю збереження всієї історії спілкування з клієнтами (дзвінки, відправка електронних листів і т.д.), автоматично виводити нагадування, за потребою виконувати ті або інші завдання, реєструвати виконання цих завдань, надати співробітникам максимально зручну інформаційне середовище для виконання своїх обов'язків.

- Формування клієнтської бази постійних і потенційних клієнтів
- Поділ клієнтів по менеджерам
- Прісвоєння клієнтам додаткових критеріїв (статус, причина звернення, реклама)
- Реєстрація звернень від потенційних клієнтів
- Занесення анкет клієнтів фітнес клубу

- Пошук в системі за основними ідентифікаторами клієнта (номер карти, моб. Телефон)
- Реєстрація менеджерами виконання завдань (дзвінки, презентації, відправка e-mail листів)
- Можливість підключення web камери (прикріплення фотографії клієнта)
- Доступ до повної інформації по клієнту і його контрактами

## РОЗДІЛ 3

# ОПИСАННЯ РОЗРОБКИ СИСТЕМИ ДЛЯ АВТОМАТИЗАЦІЇ РОБОТИ АДМІНІСТРАТОРА ТРЕНАЖЕРНОГО ЗАЛУ

### 3.1 Вибір операційної системи

Android - це операційна система (ОС) від Google, призначена для смартфонів, планшетів, наручних годинників, ігрових приставок, телевізорів та інших мультимедійних пристроїв. На сьогоднішній день Android продовжує впевнено зміцнювати свої позиції на ринку мобільних операційних систем. Це підтверджують дані різних аналітичних компаній. Так, наприклад, за даними сервісу GlobalStats частка Android на сьогоднішній день становить 74,85% [7].

Основні переваги даної ОС:

1. Android - це система з відкритим вихідним кодом, що відкриває великі можливості для розробників.
2. Кастомізація. Користувач має досить зручний і гнучкий інструментарій і з легкістю може підлаштувати інтерфейс і систему під свої потреби.
3. Великий вибір додатків в PlayMarket, а також можливість їх установки зі сторонніх ресурсів.
4. Оперативні поновлення - версії системи виходять досить часто і з кожним разом функціонал стає все краще.

Виходячи з усього перерахованого вище було прийнято рішення про реалізацію програми на базі ОС Android.

Кафедра КСМ				НАУ 21 03 46 000 ПЗ			
Виконав	Ббенко Д.С.				Літера	Аркуш	Аркушів
Керівник	Дрововозов В.І.					43	60
Консульт.					123 КС 421Б		
Н. контроль	Журавель С.В.						
Зав. Каф.	Жуков І.А.						

### 3.2 Вибір середовища розробки

Android Studio є офіційною інтегрованим середовищем розробки (IDE) програмного забезпечення на платформі Android [8]. Дана IDE була заснована на ПО від компанії JetBrains - IntelliJ IDEA. Android Studio пропонує більш просту середу розробки та налаштування проекту, і спеціально була орієнтована на створення Android-додатків. Тому, на додачу багатофункціонального редактору коду від IntelliJ, який надає широкі можливості для розробника, Android Studio містить ще більше інструментів і функцій для створення і налагодження додатків:

Редактор WYSIWYG (скор. Від англ. What You See Is What You Get - «що бачиш, то отримаєш») - призначений для роботи з макетами, рендеринга (англ. rendering - візуалізація) програми, дозволяє бачити запущене застосування на багатьох екранах і дозволах.

Можливість створення додатків для Android Wear і Android TV. Система автоматичного складання додатків Gradle.

Рефакторинг (англ. Refactoring) - це процес перетворення коду, що не міняє при цьому поведінки програмного продукту, а лише здійснює невеликі зміни його внутрішню структуру.

Lint - статичний аналізатор коду, який допомагає виявляти проблеми з продуктивністю, продуктивністю, сумісністю версій і т.д.

Наявність шаблонів для створення макетів і компонентів Android.

Керівництво користувача на офіційному сайті Android Studio - величезний плюс для початківців розробників.

Отже, для того щоб почати працювати з цією IDE, необхідно встановити пакет

JDK (скор. Java Development Kit) останньої версії. Це безкоштовний комплект розробника, призначений для створення додатків на мові Java. Знайти його можна на офіційному сайті компанії Oracle.

Тепер можна завантажувати Android Studio з офіційного сайту розробника.

Інтерфейс даної IDE представлений на Рис. 3.1.

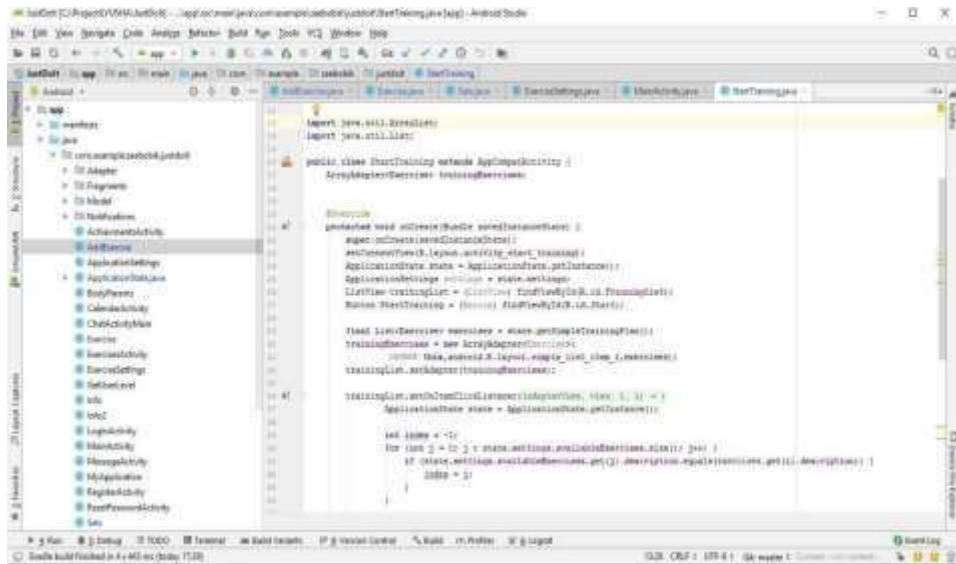


Рис. 3.1. Интерфейс Android Studio

Крім самої студії, встановляються такі інструменти розробки як Android SDK (англ. Software Development Kit) - набір засобів для розробки Android-додатків, компілює написаний розробником код в файл APK (програмний пакет Android). Містить, наприклад, такий інструмент, як AVD (англ. Android Virtual Device) [8].

AVD - це емулятор Android пристрою, який можна запускати прямо на комп'ютері. Він необхідний для проектування, налагодження і тестування розробляються, якщо під рукою немає реального смартфона на платформі Android.

AVD визначає характеристики реального пристрою - назва, діагональ екрану, версія ОС і т.п. Завдяки цьому можна створювати кілька таких віртуальних пристроїв, щоб побачити, як буде виглядати додаток на різних смартфонах.



XML (англ. EXtensible Markup Language) - це розширювана мова розмітки, синтаксис якого дуже простий і зручний для створення і подальшої обробки документів, який зрозумілий як людині, так і комп'ютера [10]. Можливість розширення полягає в тому, що XML не фіксує розмітку, а дозволяє розробнику самому створювати її в залежності від особливостей конкретної області. Обмеження при цьому складуть тільки синтаксичні правила мови. XML базується на кодуваннях Юнікод.

В Android-додатках для користувача інтерфейс часто завантажується саме з файлів з розширенням .xml, завдяки цьому відбувається відділення програмного коду,

написаного на Java від визначення інтерфейсу. Це нагадує процес створення веб-сайту, при якому визначення інтерфейсу відбувається в html-файлах, а вся логіка знаходиться в коді, написаному на JavaScript.

JSON (англ. JavaScript Object Notation) - формат обміну даними, який хоч і походить від мови JavaScript, однак при цьому може взаємодіяти практично з будь-якими мовами програмування [11]. Зручність JSON полягає в тому, що це звичайний текст, тому використовуючи його, можна з легкістю виробляти передачу і обмін даних між різними мовами програмування, БД і серверами.

Дані в форматі JSON можуть являти собою:

Об'єкт. Це неупорядкований набір пар «ключ: значення». Полягає в фігурні дужки.

Одновимірний масив. У квадратних дужках.

Рядок. Полягає в подвійних лапках. Число.

Літерали (true, false, null).

### **3.2. Реалізація системи**

Android-додатки в Android Studio складаються з наступних компонентів [8]:

1. Activity (діяльність, активність) - це компонент додатка, вдає із себе екран для взаємодії з користувачем.

2. Service (служба) - компонент, який працює у фоновому режимі і не має призначеного для користувача інтерфейсу.

3. Content-provider (контент-провайдер) - це «постачальник контенту», управляє даними в додатку.

4. Broadcast Receiver (широкомовний приймач повідомлень) - це компонент додатка, що реагує на оголошення, які поширюються по системі.

5. Intent (Інтент, намір) - це асинхронне повідомлення, яке передається між компонентами і визначає наміри користувача виконати будь-яку дію.

Бувають явні (для виклику конкретного компонента) і неявні (коли невідомо, який саме компонент повинен бути запущений) наміри.

6. Layout (макет) - це визначення візуальної структури призначеного для інтерфейсу користувача.

Розглянемо докладніше такий важливий компонент, як Activity, без якого не обійдеться жодне додаток на Android.

Отже, як вже було сказано раніше, Activity - це один з компонентів Android-додатки. Зазвичай програма містить кілька Activity, тобто екранів, які повинні взаємодіяти один з одним.

Activity, що запускається найпершою, вважається головною, з неї вже здійснюються переходи в інші Activity. У міру того, як користувач переходить з одного екрану в інший, Activity знаходяться на різних етапах свого життєвого циклу (Рис. 15).

Як видно на схемі, Activity може мати три стани:

1. Активна (active, running) - стан, при якому Activity виконується, тобто відображається користувачеві на екрані.

2. Призупинена (paused) - на передньому плані екрана відображається інша активність, проте дана Activity все ще не прихована від користувача і знаходиться в пам'яті.



Система може завершити її в тому випадку, якщо буде необхідна вільна пам'ять.

3. Зупинена (stopped) - стан, коли Activity повністю закривається інший Activity і стає невидимою для користувача.

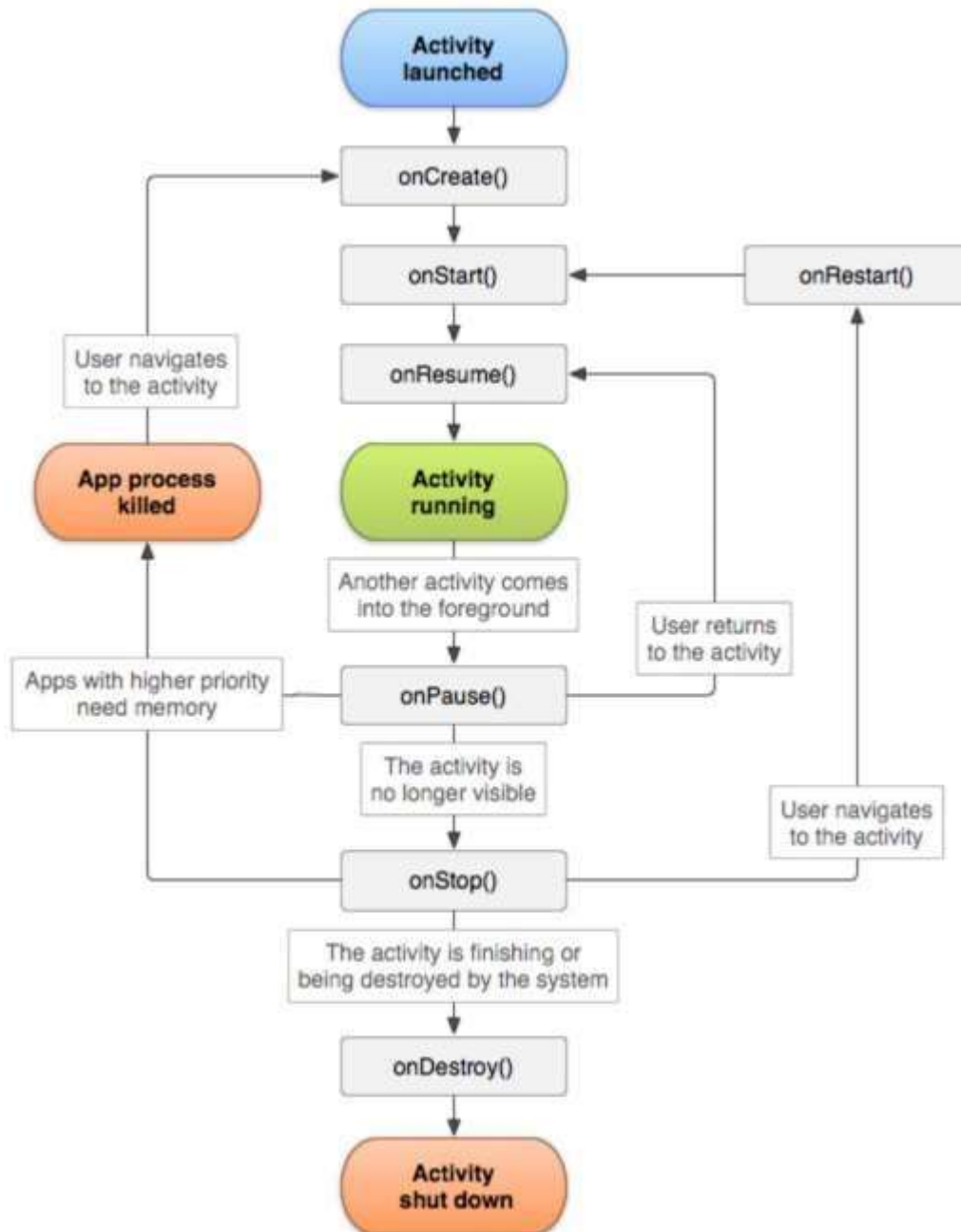


Рис. 3.3. Життєвий цикл Activity

При переході Activity з одного стану в інший викликається один з методів [8]:

1. `onCreate ()` - це найперший метод, з нього Activity починає своє виконання.

Його необхідно визначати в класі Activity. Даному методу передається об'єкт Bundle, що містить попередній стан Activity, якщо таке було збережено, або значення null, якщо Activity створюється заново.

2. `onStart ()` - це метод, що викликається перед тим, як Activity стане видимою для користувача.

3. `onResume ()` - при виклику даного методу користувач може безпосередньо взаємодіяти з Activity, тобто вона знаходиться у фокусі - до тих пір, поки користувач не переключиться на іншу Activity, або поки не згасне екран мобільного пристрою.

4. `onPause ()` - викликається при переході користувача в іншу Activity. Activity

після виклику даного методу не відображається на екрані, проте ще залишається активною. Для того щоб повернутися в Activity, викликається метод `onResume ()` і вона знову стає видимою.

5. `onStop ()` - викликається тоді, коли Activity більше не видно для користувача,

проте вона ще зберігається в пам'яті пристрою.

6. `onDestroy ()` - даним методом завершується Activity.

7. `onRestart ()` - це метод, що викликається при перезапуску Activity.

Всі перераховані методи можуть бути перевизначені (override), для того щоб виконувати необхідну роботу при зміні стану Activity.

MVC (англ. Model-View-Controller) - це концепція програмної архітектури, призначена для відділення логіки додатка від призначеного для користувача інтерфейсу [12].

Дана концепція передбачає виділення трьох окремих компонентів: моделі, уявлення і контролера таким чином, щоб модифікація одного з них надавала мінімальний вплив на інші компоненти.

Під моделлю в MVC маються на увазі класи, які відповідають за зберігання даних і правила роботи з ними. При цьому модель не повинна знати нічого про те, як саме ці дані будуть відображатися.

Подання відповідає за користувальницький інтерфейс програми. В Android це макети, тобто layout-и, вбудовані класи, наприклад, Button і т.п. Зазвичай layout рекомендовано створювати у вигляді окремих xml-файлів в папці ресурсів, тим самим виконується головна умова концепції MVC - відділення логіки від інтерфейсу.

З контролером все просто - в Android це Activity або Fragment.

Таким чином, користувач взаємодіє з Controller-му, який, в свою чергу, запитує у Model дані і заповнює їх у View, який безпосередньо відображається користувачеві.

Singleton (англ. - одиночка) - це породжує патерн, який заснований на ідеї використання єдиного примірника деякого класу, до якого надається глобальна точка доступу [13].

До цього примірника надається глобальна точка доступу - метод getInstance (). Дана властивість допомагає, наприклад, коли об'єкт класу працює з даними, і необхідно здійснювати доступ до даних з будь-яких частин програми. Тим самим Singleton гарантує, що раніше створений instance ні замінений ніяким іншим кодом.

### **3.3. Компоненти мобільного додатка**

Для того щоб згрупувати класи в компоненти більш високого рівня, в UML використовуються так звані пакети (англ. Package). А залежності між ними відображає діаграма пакетів [14]. На Рис. 3.4 зображено саме така діаграма для розробляється.

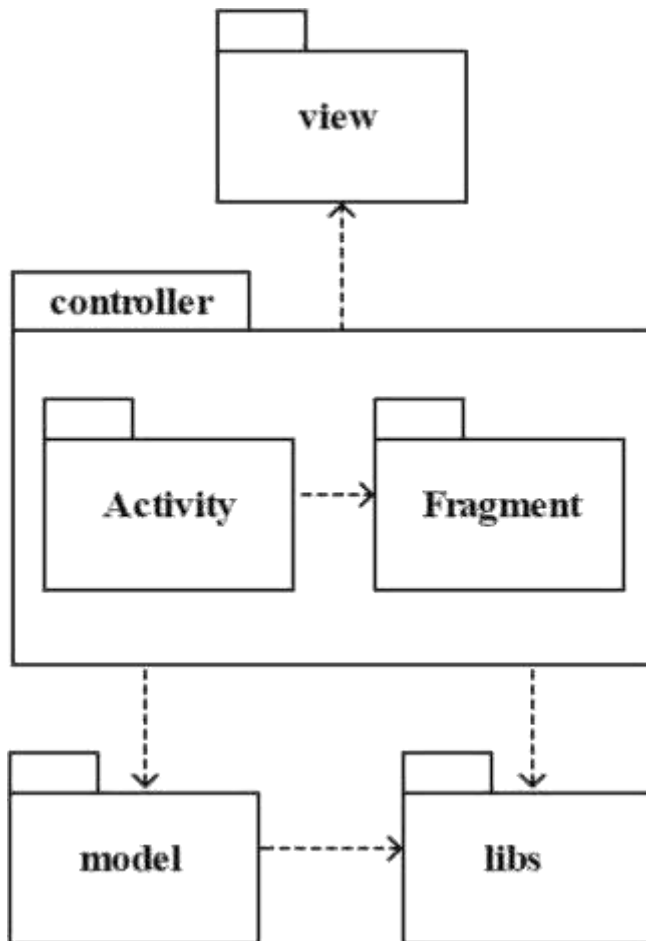


Рис. 3.4. Діаграма пакетів

На діаграмі представлено розподіл даних програми згідно з концепцією патерну MVC, про який вже говорилося трохи раніше.

У пакеті `view` розташовуються файли з папки `app / res`, а саме: всі макети (layout), зображення, стилі і т.п.

Пакет `libs` містить бібліотеки, які використовуються при створенні програми.

Всі класи моделі розташовуються відповідно в пакеті `model`. Це такі класи, як `ApplicationState`, `ApplicationSettings` (рисунок 18), необхідні для складання плану тренувань, а також класи `User`, `Chat`, `Chatlist`, які потрібні для роботи месенджера.

В пакеті `controller` містяться всі `Activity`, а також фрагменти (Fragments) - по суті це аналог `Activity`, який можна використовувати в різних частинах програми.

На Рис. 19 зображується діаграма класів контролерів.

Як можна помітити, на діаграмі відображені можливі переходи між екранами додатка.

Опис контролерів:

MainActivity - головний екран додатка, через який здійснюються переходи в інші основні компоненти системи;

InfoActivity - вітальний екран, запускається при найпершому запуску програми;

BodyParamsActivity - екран введення призначених для користувача параметрів, необхідних для складання тренування;

StartTrainingActivity - екран, необхідний для відображення тренування;

TrainingActivity - необхідний для виконання безпосередньо тренування;

CongratsActivity - додатковий екран, відображається після завершення тренування;

ExerciseSettingsActivity - для зміни налаштувань конкретного вправи (Наприклад, кількості підходів, повторень або ваги);

AddExTrainingActivity - для додавання нового вправи в тренування;

SettingsActivity - необхідний для змін налаштувань і параметрів користувача;

AchievementsActivity - необхідний для відображення досягнень користувача; ExercisesActivity - для реалізації каталогу вправ;

AddExerciseActivity - необхідний для додавання нового вправи в каталог;

ShowExerciseActivity - для відображення опису конкретного вправи;

CalendarActivity - для відображення календаря тренувань;

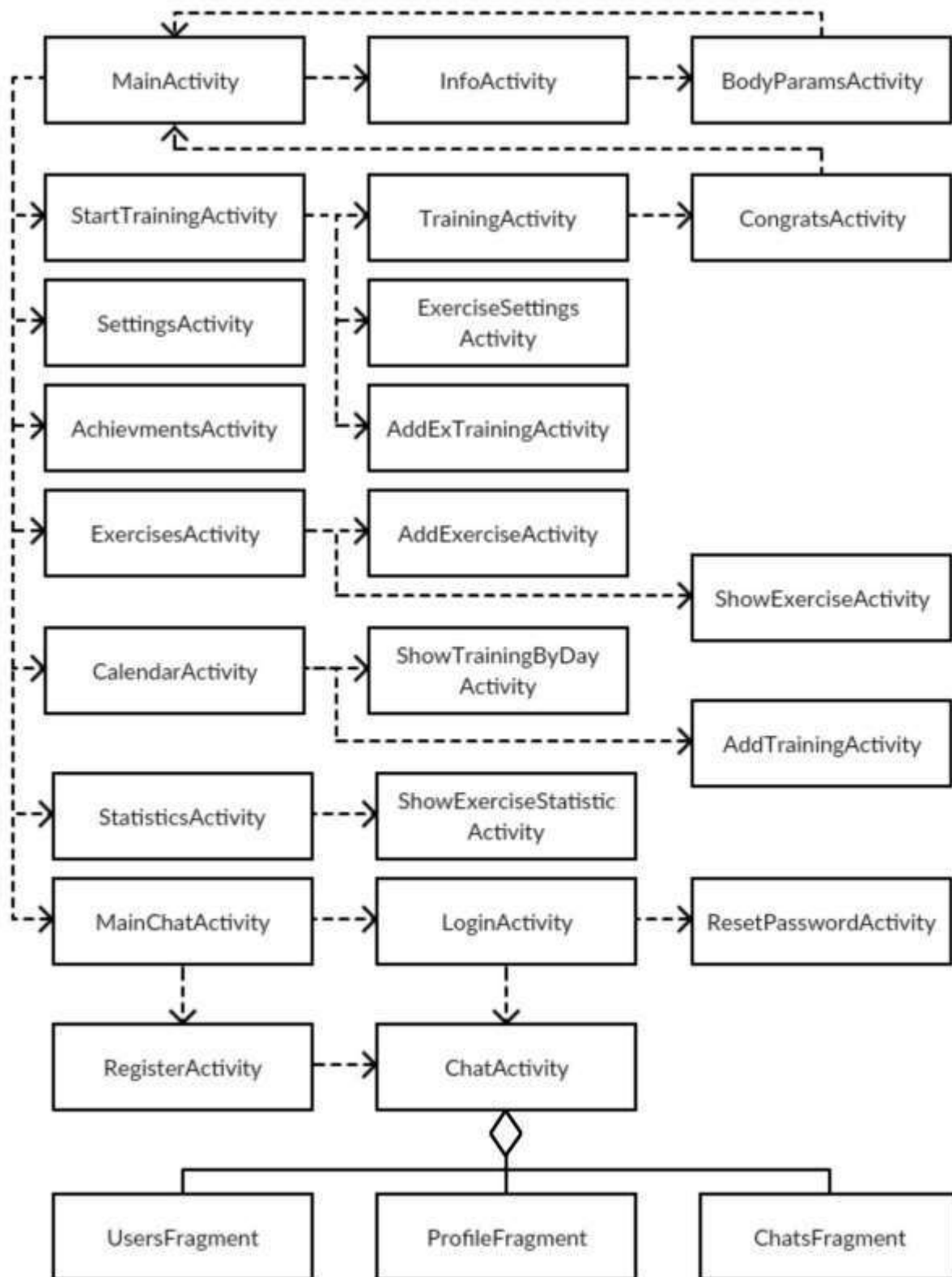


Рис. 3.5. Діаграма класів контролерів

ShowTrainingByDayActivity - необхідний для відображення тренування в конкретний день (минулої або запланованої);

AddTrainingActivity - необхідний для планування тренування;  
StatisticsActivity - відображення статистики користувача;

ShowExerciseStatisticActivity - відображення статистики по конкретному вправі;

MainChatActivity - головний екран для входу в месенджер;

LoginActivity - для авторизації користувача в месенджері; RegisterActivity - для реєстрації користувача в месенджері; ResetPasswordActivity - для відновлення пароля;

ChatActivity - головний екран месенджера, містить три вкладки, реалізовані за допомогою Fragments;

UsersFragment - список користувачів месенджера;

ProfileFragment - щоб зробити режим користувача в месенджері;  
ChatsFragment - відображення листувань користувача.

В додатку, що розробляється дані зберігаються в форматі JSON. Постає питання про те, як же працювати з цими даними, як переводити деякий java-об'єкт в JSON уявлення і навпаки.

Розглянемо бібліотеку Gson, призначену саме для цих цілей, тобто для серіалізації і десеріалізації java-об'єктів в JSON і назад.

Дана бібліотека була розроблена програмістами Google і спочатку використовувалася всередині компанії, проте пізніше бібліотеку вирішили віддати в open-source для подальшого її розвитку.

Бібліотека Gson як і всі інші бібліотеки підключається в файлі gradle рівня app. Вона гранично проста у використанні і розумінні. Все, що необхідно знати, це два методи:

toJson () - для перетворення java-об'єктів в JSON уявлення; fromJson () - для перетворення JSON уявлення в java-об'єкти.

### **3.4. Внутрішня реалізація складання тренувань**

Вхідними даними є параметри користувача, введені при першому запуску програми або при зміні в настройках, а саме:

вік;  
зріст;  
вага;  
стать;  
мета занять;  
наявність досвіду в тренуваннях;  
бажану кількість тренувань в тиждень.

Для складання тренувань на тиждень вихідні дані по кількостях вправ, повторів, підходів, часом відпочинку і т.д. зберігаються в json-файлі. Для різних цілей ці дані різняться.

На основі цих показників за допомогою методу `generateNewTraining ()` генеруються тренування. У чергу вправ `trainingQueue` додаються тільки ті, які підходять при конкретній меті тренувань. Для схуднення вправи беруться з усіх груп м'язів, як і для підтримки форми, для набору маси вибираються вправи, для яких вказана відповідна мета (`target`).

Метод тренування (кругової / роздільний) впливає на те, в якому порядку при виконанні будуть відображатися підходи. Для першого варіанту цей порядок буде наступним: перший підхід першої вправи, потім відпочинок (включається таймер), потім перший підхід другої вправи і т.д.

Для роздільного тренування реалізований наступний принцип: спочатку перший підхід першої вправи, відпочинок, другий підхід першої вправи і т.д.

Для такої реалізації необхідний додатковий клас `Sets` - клас підходів. Після завершення тренування, дані про виконані вправах записуються в `HashMap trainingHistory` і зберігаються в файл `global_settings`. Надалі ці дані використовуються для відображення статистики по вправам (`showExerciseStatistic`) і для відображення виконаної тренування в календарі (`doneSetsOfCurrentTraining`).

При натисканні на «Мессенджер» відкривається вікно для входу і реєстрації (рисунк 37). Екран входу (рисунк 38) містить поля для введення даних (e-mail і пароль).



На рис. 3.7 представлена модель авторизації користувачів, на рис. 3.8 – процес придбання користувачем абонементу, на рис. 3.9 – формування звітності. Ці моделі описують не всі процеси в системі, але є основними.

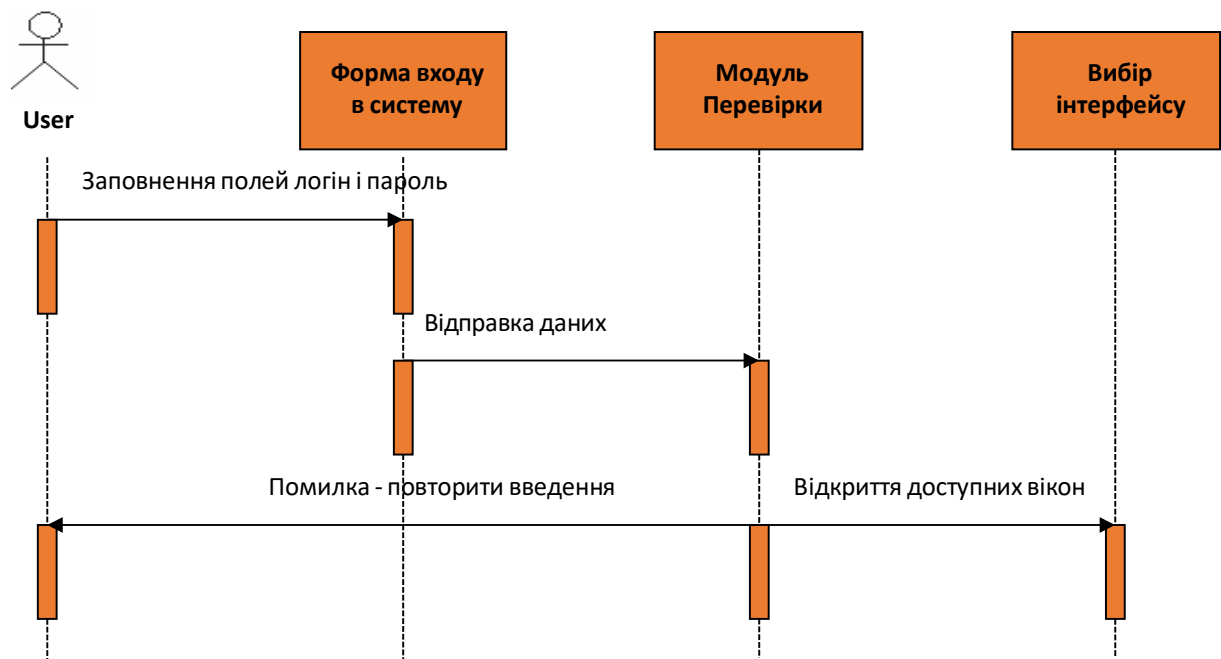


Рис. 3.1. Діаграма взаємодії з системою на етапі авторизації

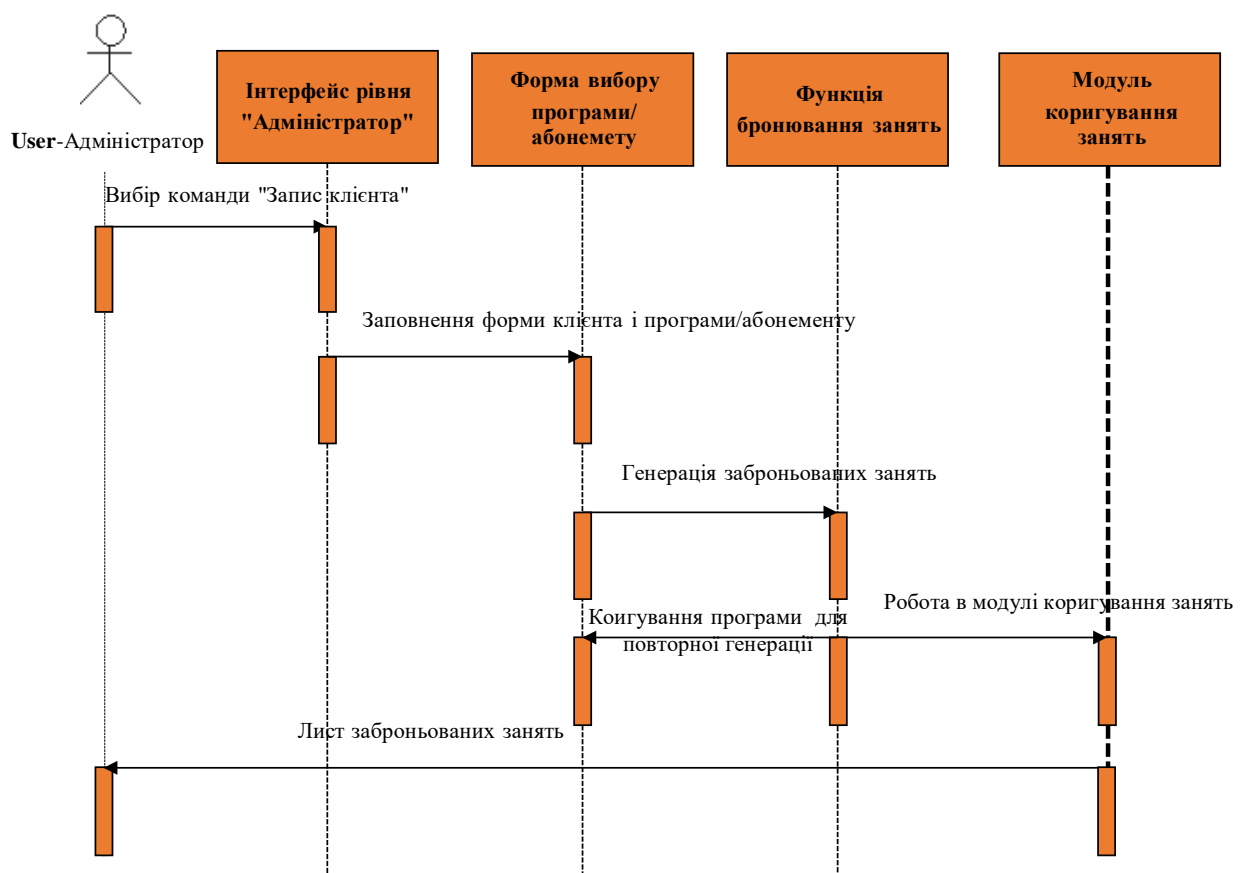


Рис. 3.2. Діаграма взаємодії з системою на етапі продажу абонементу

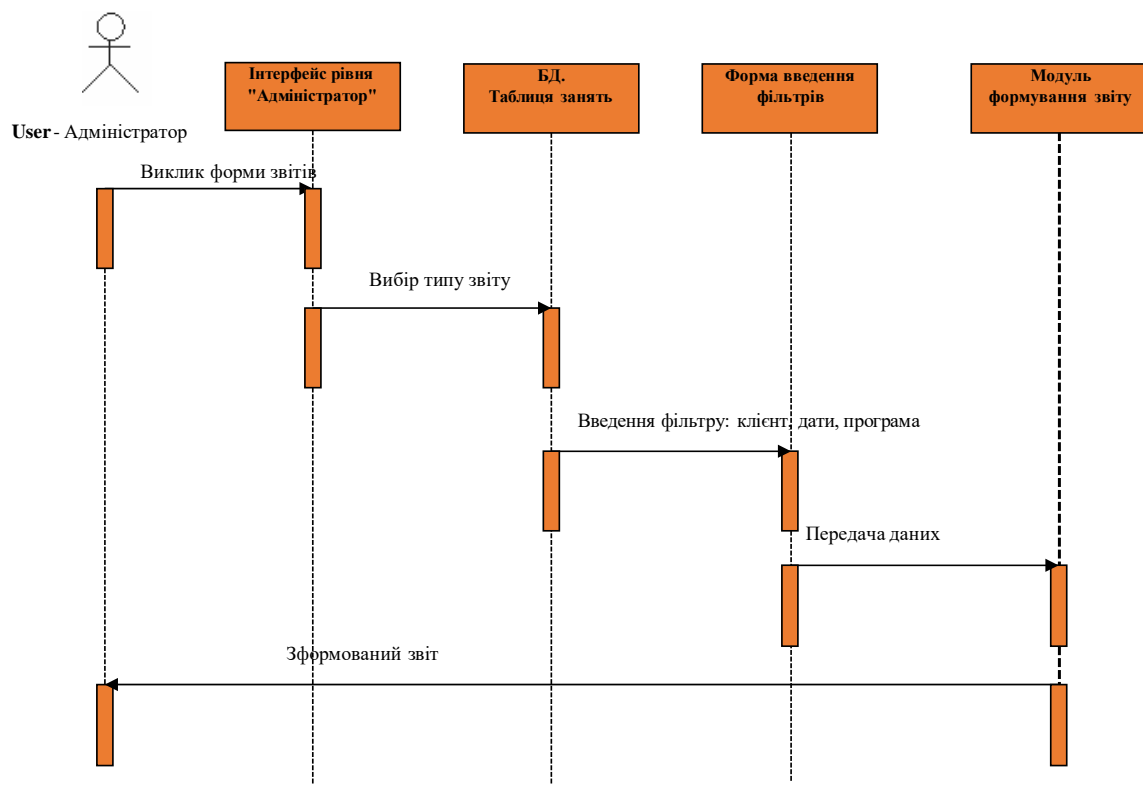


Рис. 3.3. Діаграма взаємодії з системою на етапі формування звітності

**Бронирование мест на занятиях**

Существующие занятия | Создать новое занятие

На дату:  Филлиал:  Группы:   
 На неделю:  Направление:  Зан:   
 Посетитель:     
 Преподаватель:

**Занятия, доступные для бронирования**

ДН Дата	Занятие	Филиал-Зан	Свободно / Бронь / Всего	Бронь	Ссылка
Пн 06.05.2019.	09:00 - 10:00 Дет фит - Дет фит 3-5 лет Александр М	Филиал на Печерске - Зан №1 (Большой)	4 / 0 / 4		
Пн 06.05.2019.	10:00 - 11:00 Пилатес - Раша прогр Илья П Ольга К	Филиал на Печерске - Зан №1 (Большой)	4 / 0 / 4		
Пн 06.05.2019.	11:00 - 12:00 Фитнес группа старшие - Фитнес Марина К	Филиал на Печерске - Зан №1 (Большой)	4 / 0 / 4		

Рис. 3.4. Форма бронювання занять

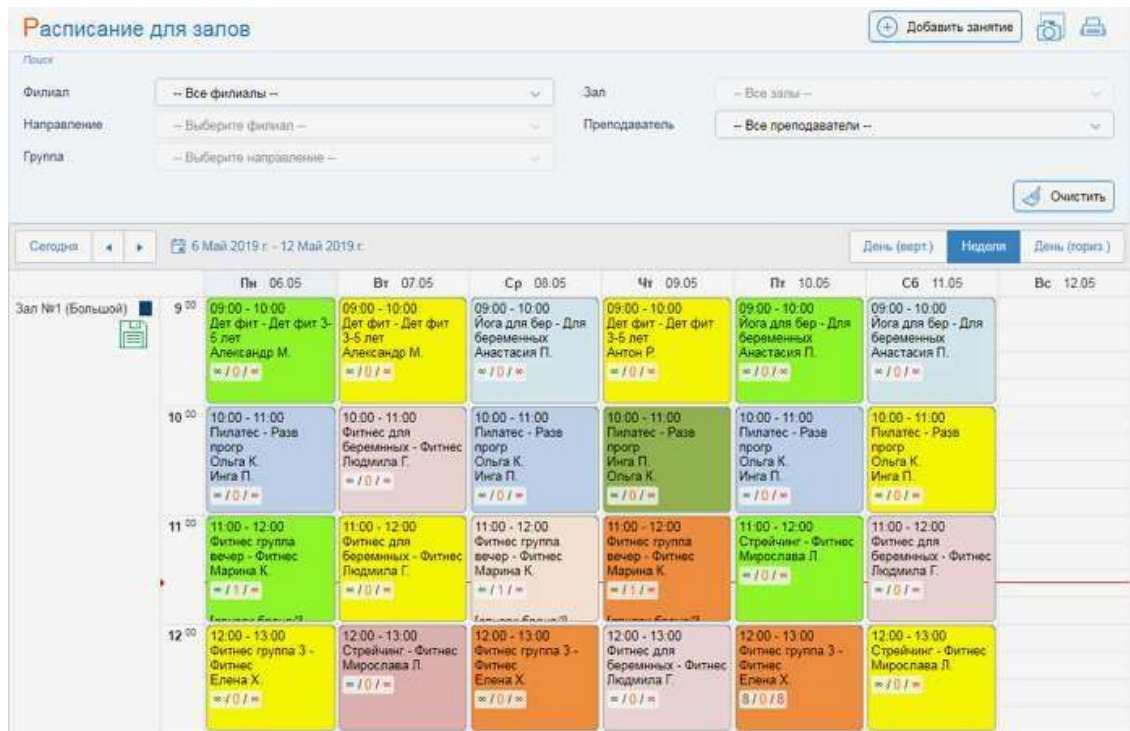


Рис. 3.5. Форма зайнятості залів



Рис. 3.7. Форма карты клієнта

Приход товаров

От поставщика:  На склад:

Накладная №:  от:

Артикул	Товар	Упаковки	Количество	На складе	Сумма	Валюта
	Вода и соки	50				
	Bonaqua 0.5л	10	5,00000 л	2,0 шт. (1,00000 л)	300,00	\$
	Evian 0.5л	10	5,00000 л	0,0 шт. (0,00000 л)	290,00	\$
	Perrier 0.33л	10	3,30000 л	0,0 шт. (0,00000 л)	3 470,00	\$
	Rich 1л	10	10,00000 л	0,0 шт. (0,00000 л)	4 680,00	\$
	Vittel 0.33л	10	3,30000 л	0,0 шт. (0,00000 л)	1 780,00	\$
	Кофе	19				
	Американо	0	0 шт.	7,0 шт. (7 шт.)	0,00	\$
	Капучино	8	8 шт.	2,0 шт. (2 шт.)	3 728,00	\$
	Латте	11	11 шт.	1,0 шт. (1 шт.)	1 210,00	\$
	Массаж	40				
	Воск для массажа 1кг	20	20,00000 кг	0,0 шт. (-0,00300 кг)	7 300,00	\$
	Масло для массажа 500мл	20	10 000,000 мл	0,2 шт. (92,000 мл)	9 180,00	\$
	Спортивное питание	30				
	Amino	10	10 шт.	1,0 шт. (1 шт.)	2 870,00	\$
	BCAA	10	10 шт.	1,0 шт. (1 шт.)	4 790,00	\$
	Протеиновый батончик	10	10 шт.	1,0 шт. (1 шт.)	1 480,00	\$

Поиск:  Фильтр Внести Excel

Общая сумма: **\$41 078,00 (0,00 грн.)**

Скидка:  % Транспортные расходы:  \$

Оплата: 

- Касса
- Банк
- Сейф
- Оплатить позже
- Под реализацию

Примечания:

☐ Изменить цены продажи

OK Отмена

Рис. 3.8. Форма приходу товарів

## Висновки до розділу

В розділі було виконано поставлені завдання, а саме:

- вивчена і проаналізована предметна область: основні принципи складання тренувань, виходячи з індивідуальних особливостей людини;
- сформульовані функціональні і нефункціональні вимоги до системи;
- вивчені і придбані навички з розробки Android-додатків на мові Java.

## ВИСНОВКИ

Внаслідок виконання випускної кваліфікаційної роботи були проаналізовані найбільш поширені програми для складання та виконання тренувань. У зв'язку з тим, що здоровий спосіб життя, в тому числі і спорт, в сучасному світі стає досить популярним, різних програмних рішень в даній сфері дуже багато. Однак, в ході аналізу було з'ясовано, що всі вони мають ряд недоліків.

Крім цього, були виконані і інші поставлені завдання, а саме:

- вивчена і проаналізована предметна область: основні принципи складання тренувань, виходячи з індивідуальних особливостей людини;
- сформульовані функціональні і нефункціональні вимоги до системи;
- вивчені і придбані навички з розробки Android-додатків на мові Java.

В підсумку було розроблено додаток на платформі Android для складання індивідуальних програм тренувань з можливістю внесення даних про їх виконання, яке усунуло недоліки аналогічних програмних рішень, і при цьому об'єднало їх гідності. Тим самим була досягнута поставлена мета даної роботи.

Надалі планується поліпшити інтерфейс користувача, а також розширити функціонал додатка, додавши можливість складання плану харчування. Адже ведення здорового способу життя - це не тільки виконання тренувань, але і дотримання режиму харчування.

Важливо розуміти, що повністю замінити тренера для цієї програми не зможе, воно носить лише рекомендаційний характер і допомагає полегшити життя спортсменам.

Використання основних можливостей програми для фітнес клубу дозволяє досягати таких цілей:

- Автоматизація роботи рецепції клубу: реєстрація відвідувань клієнтів, перевірка активності карт, продаж товарів і послуг

- Ведення фінансових взаєморозрахунків, організація безготівкового розрахунку всередині фітнес клубу
- Впровадження CRM системи для відділу продажів: облік дзвінків, друк договорів, продаж контрактів, реєстрація завдань
- Форма для роботи тренерського складу - запис і списання персональних тренувань, надання звітів
- Удосконалювати методи реклами на базі індивідуальних SMS і e-mail розсилок
- Оптимізувати облік і контроль витрат і доходів, включаючи складської моніторинг витратних матеріалів та препаратів
- Удосконалювати кадрову роботу - скласти оптимальні робочі графіки, проводити моніторинг економічної ефективності співробітників, нараховувати заробітну плату
- Сформувати єдиний автоматизований комплекс, на основі інтегрованого в програму управління контрольно-облікового обладнання
- Створити багаторівневу систему безпеки, на основі систем контролю і управління доступом (СКУД)

Основне завдання вбудованого CRM модуля для фітнесу - це надання адміністраторам і менеджерам відділу продажів фітнес клубу таких механізмів, за допомогою яких вони зможуть оперативно і своєчасно виконувати свої обов'язки. Якщо говорити про фітнес індустрії, то основні функції, які повинна виконувати CRM система - це ведення клієнтської бази з можливістю збереження всієї історії спілкування з клієнтами (дзвінки, відправка електронних листів і т.д.), автоматично виводити нагадування, за потребою виконувати ті або інші завдання, реєструвати виконання цих завдань, надати співробітникам максимально зручну інформаційне середовище для виконання своїх обов'язків.

- Формування клієнтської бази постійних і потенційних клієнтів
- Поділ клієнтів по менеджерам
- Прісвоєння клієнтам додаткових критеріїв (статус, причина звернення, реклама)

- Реєстрація звернень від потенційних клієнтів
- Занесення анкет клієнтів фітнес клубу
- Пошук в системі за основними ідентифікаторами клієнта (номер карти, моб. Телефон)
- Реєстрація менеджерами виконання завдань (дзвінки, презентації, відправка e-mail листів)
- Можливість підключення web камери (прикріплення фотографії клієнта)
- Доступ до повної інформації по клієнту і його контрактами

Реалізовану в межах дипломного проекту систему можна впроваджувати в будь-які тренажерному залі.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ДСТУ 3008-95. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення. – 39 с.
2. Бойченко С.В., Іванченко О.В. Положення про дипломні роботи (проекти) випускників Національного авіаційного університету / С.В. Бойченко, О.В. Іванченко – К.: НАУ, 2017. – 63 с.
3. Akyildiz I.F. *Wireless multimedia sensor networks: applications and testbeds* / Akyildiz I. F., Melodia T., Chowdury K.R. // *Proceedings of the IEEE (invited paper)*, 2008. – Vol. 96. – № 10 – Pp. 1588-1605.
4. Akyildiz I.F., Vuran M.C. *Wireless Sensor Networks* / I.F. Akyildiz, M.C. Vuran – New York: John Wiley & Sons, 2010. – 571 p.
5. Basagni S. *Mobility – Adaptive Protocols for Managing Large AD Hoc Network* / S. Basagni, D. Turgut, S.K. Das // *Proceedings of the IEEE International Conference on Communications (ICC)*. – 2011.-P. 1539-1543.
6. Broch, J. *A performance comparison of multihop wireless ad hoc network routing protocols* / J. Broch, D.A. Maltz, D.B. Johnson, J. Jetcheva // *Proc. of MOBICOM'08*. – 2008.
7. Kawadia V. *System services for implementing Ad-Hoc routing: Architecture, Implementation and Experiences*/ V. Kawadia, Y. Zhang, B. Gupta // *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services (MobiSys)*. San Francisco, CA. – June 2013. – P. 99-112.
8. Mockapetris. *P Domain Names – Concepts and Facilities*. – Network Working Group. – 2017.
9. Perkins, C.E. *Ad hoc On-Demand Distance Vector (AODV) Routing* / C.E. Perkins, C.E. BeldingRoyer // *RFC 3561*. – July 2013.
10. Schulman A. *The Extent of Systematic Monitoring of Employee E-mail and Internet Use, US, 2010-2012*. Постійне посилання: <http://www.sonic.net>
11. Shin, K.G. *Performance Analysis of Distributed Routing Strategies Free of Ping-Pong-Type Looping* / K.G. Shin, M. Chen // *IEEE Trans. Computers*. – February 2007. – V. COMP-36, M 2. – P. 129-137.
12. Бодрова А. А., Логвин В.И. *Позиционирование и взаимодействие в беспроводных сенсорных сетях* / А.А. Бодрова, В.И. Логвин // *Молодой ученый*. –



13. Болілий В.О., Котяк В.В. Комп'ютерні мережі. Навчальний посібник / В.О. Болілий, В.В. Котяк – Кіровоград: ЦОП Авангард, 2008.– 146 с.

14. Власенко В.О. Методи самоорганізації безпроводних сенсорних мереж // Сучасний захист інформації №1, 2017. – С. 96-102.

15. Вороненко А.В., Романов В.А. Выбор унифицированных средств процессорной обработки данных на основе *cortex*-ядра для биосенсорных приборов // Комп'ютерні засоби, мережі та системи: зб. наук. пр. / К.: Ін-т кібернетики ім. В. М. Глушкова НАН України. – 2013 (№12). – С. 44-101.

16. Высокопроизводительные сети. Энциклопедия пользователя./ Под ред. Марка А. Спортака: Пер. с англ. – К.: ДияСофт, 2008. – 836 с.

17. Груша В.М., Артеменко Д.М., Пацко О.В. Використання бездротового зв'язку для моніторингу стану насаджень методом індукції флуоресценції хлорофілу / В.М. Груша, Д.М. Артеменко, О.В. Пацко // XVIII Міжнародна конференція з автоматичного управління, 28-30 вересня 2011 року: мат. конф. – Львів: Видавництво Львівської політехніки. – 2011. – С. 392-393.

18. Коцюр А.Б., Казадаєв І.О. Підсистема теледіагностики на основі бездротових технологій / А.Б. Коцюр, І.О. Казадаєв // Тези доповідей наук.-практ. конф. “Сучасні тенденції розвитку системного програмування” (25-26 листопада 2020 р.). – К.: НАУ, 2020. – С. 42.

19. Коробейников А.В. Структура системы мониторинга для медицинских учреждений / Математическое моделирование и интеллектуальные системы. – Ижевск: ИжГТУ, 2003. – № 1. – С. 18-21.

## ДОДАТОК А

### Схема алгоритму формування звіту

