

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук,
управління та адміністрування
Кафедра інформаційних технологій

Бакалаврська кваліфікаційна робота

на тему: Розробка IoT-проекту метеостанції

Виконав студент 4 курсу групи К-41
Напрям 6.050101 2 комп'ютерні науки,
Ганев Василь Олексійович

Керівник асистент
Штефан Наталія Зінов'ївна

Консультант: к.т.н., доцент
Великодний Станіслав Сергійович

Рецензент: к.георг.н., доцент
Бояринцев Євген Львович

ЗМІСТ

Скорочення та умовні позначки	8
Вступ.....	10
1 Аналітичний розділ	12
1.1 Опис предметної області	12
1.2 Характеристика об'єкту розробки.....	13
1.3 Аналіз існуючих аналогів.....	14
1.4 Вибір програмних засобів розробки	20
1.4.1 Мови програмування C# і Java	20
1.4.2 Середовище розробки Microsoft Visual Studio	23
1.4.3 Середовище розробки для Java IntelliJ IDEA	25
1.4.4 Інструментарій JavaFX	27
1.4.5 Технологія .Net Core	28
1.4.6 Система управління базами даних MySQL	29
2 Проектна частина	30
2.1 Проектування інтерфейсу чатбота	30
2.1.1 Правила Юзабіліті для інтерфейсів.....	30
2.1.2 Проектування макетів для форм.....	32
2.2 Алгоритми роботи чатбота.....	35
2.2.1 Алгоритм Дамерау — Левенштейн.....	35
2.2.2 Діаграма Use-Case для контролера запитів	36
2.2.3 Діаграма діяльності для алгоритму роботи бота	38
2.3 Розробка структури бази даних	41
3 Реалізація проекту.....	42
3.1 Сервер боту.....	42
3.1.1 Telegram API	42
3.1.2 Функціонал боту та алгоритм Дамерау Левенштейн.....	43
3.1.3 Контролер запитів	43

3.2 Опис інтерфейсу бота	44
3.2.1 Вікно «Сервера»	44
3.2.2 Головне меню	45
3.2.3 Вікно «Контакти».....	46
3.2.3 Вікно «Види танців»	47
3.2.4 Вікно «Возрасные группы»	48
3.2.5 Вікна «Новий запис».....	50
3.2.6 Вікно «Рекламний контент».....	50
Висновки	52
Перелік джерел посилання	53

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

CSS – Cascading Style Sheets

HTML – HyperText Markup Language

Java EE – Java Platform, Enterprise Edition

JVM – Java Virtual Machine

POSIX – Portable Operating System Interface

XML – Extensible Markup Language

Аплети – прикладна програма, найчастіше написана на мові програмування Java в формі байт-коду

Вебвью – це компонент платформи Android, який дозволяє вбудовувати web-сторінки в Android-додатки

Рефакторизація – процес переробки коду з метою підвищення його удобочитаємості і продуктивності

Юзабіліті – це міра якості користувацького досвіду, набутого при взаємодії з продуктом або системою

Чатбот – штучний розум, програма-посередник

Android – операційна система

ASP.NET – платформа розробки веб-додатків

C – мова програмування

C ++ – мова програмування

IDE-середовища Visual Studio – це набір інструментів і засобів розробки різного роду застосувань

Fuzzy string search – алгоритми нечіткого пошуку

Java – мова програмування

Java Runtime – виконавче середовище для запуску і роботи додатків

JavaScript – повноцінний динамічний мову програмування, який застосовується до HTML документу, і може забезпечити динамічну інтерактивність на веб-сайтах

Grails – фреймворк для створення веб-додатків, написаний на скриптовій мовою Groovy, який в свою чергу заснований на Java

Microsoft Visual Studio – це інтегроване середовище розробки

Telegram – месенджер

Top Stories – послідовність новин у месенжерах

Spring Framework – універсальний фреймворк з відкритим вихідним кодом для Java-платформи

.NET Core – технологію від компанії Microsoft Facebook Messenger

ВСТУП

В міру того, як спілкування з комп'ютерами стає все більш очевидним, зростає роль діалогового спілкування природною мовою. Наприклад, можливість просити голосового помічника поставити час на будильник, замовити піцу, забронювати готель, купити квитки і т.д.

Все це може бути вирішено за допомогою діалогу. Більш того, це може бути не тільки діалог між користувачем і комп'ютером: в більш складному випадку людина може попросити бота спланувати подорож, і потім бот буде сам спілкуватися з іншими ботами, здійснюючи замовлення готелю та квитків [1]¹⁾.

Чатбот – штучний розум, програма-посередник, головне призначення якої – в текстовому або голосовому діалоговому режимі допомогти користувачеві з пошуком інформації, вибором товарів і послуг магазину або сервісу, виконанням нескладного дії (на кшталт бронювання столика або записи до лікаря) або просто розважити за допомогою спілкування. Як стає зрозуміло з такого визначення, сфера застосування чатботів досить широка [2]²⁾.

При цьому з високим ступенем ймовірності можна припустити, що їх швидке поширення і настільки широке використання в бізнес-сфері головним чином пов'язані з рідкісним збігом інтересів власників різних сайтів або сервісів і їх клієнтів.

Чатбот – це результат взаємодії людини зі штучним інтелектом: робот, який автоматично відповідає на повідомлення, введені користувачем в чаті. Чатбот можуть використовуватися для безлічі цілей: від виконання повсякденних завдань до розваг, а розміщують їх в месенджерах і соціальних мережах, – Facebook Messenger, Telegram і т.д. Але найчастіше саме на

¹⁾ [1] Чат-боты – следующее поколение приложений. URL: <https://habr.com/ru/company/microsoft/blog/281459/>. (дата звернення 10.03.2019).

²⁾ [2] Чат-боты: электронный умный собеседник. URL: <https://robo-sapiens.ru/stati/chat-botyi/>. (дата звернення 10.03.2019).

перших двох платформах [3]¹⁾. Їх використання, з одного боку, корисно для клієнтів, тому що вони можуть отримати потрібну інформацію або вчиняти будь-які дії більш простим і зручним способом, і, з іншого боку, це вигідно для компаній, які за допомогою чатбот можуть просунути свій бренд, зібрати від цільової аудиторії необхідні відомості і підвищити її лояльність.

Мета роботи: розробити чатбота для школи танців. Загальні характеристики кваліфікаційної роботи:

- повний обсяг сторінок пояснювальної записки – 51
- кількість рисунків – 28
- кількість таблиць – 0
- кількість посилань – 17

¹⁾ [3] Почему чатботы захватывают мир + 25 примеров для вдохновения. URL: <https://texterra.ru/blog/pochemu-chatboty-zakhvatyvayut-mir-25-primerov-dlya-vdokhnoveniya.html>. (дата звернення 12.03.2019).

1 АНАЛІТИЧНИЙ РОЗДІЛ

1.1 Опис предметної області

Сьогодні вже наявність чатбота, прокачаного віртуального співрозмовника в месенджері не так модний тренд, скільки *must have* в digital-бізнесі. Використання функціоналу чатбота є дуже зручним. Вони без заперечень виконують будь-яку рутинну задачу, пов'язану з комунікацією.

Щоб вони змогли працювати по «спеціальності», в них треба встановити відповідні алгоритми та програми. Їх кількість і можливості фактично визначаються замовником з урахуванням області застосування і майбутніми завданнями, проте в цілому функції і професійні обов'язки зумовлює принцип роботи чат-бота, однаковий для всіх подібних рішень [2]¹⁾.

Все починається з прийому повідомлення і його семантичного аналізу (виокремлення ключових слів і фраз). У відповідь на ці слова і фрази запускається побудова діалогу відповідно до закладених в софт-сценаріями. Їх деталізація, інформаційне наповнення і кінцеві цілі і визначають кваліфікацію бота в якості віртуального співрозмовника [4]²⁾.

Крім стандартного подання інформації, боти можуть на повному автоматі виконувати такі задачі, як:

- передавати контакти в сервіс розсилки;
- надавати технічну підтримку;
- консультувати клієнтів, відповідати на питання;
- проводити опитування і тести;
- збирати і запам'ятовувати інформацію про клієнта;
- робити розсилку корисного контенту та рекламних матеріалів;

¹⁾ [2] Чат-боты: электронный умный собеседник. URL: <https://robo-sapiens.ru/stati/chat-botyi/>. (дата звернення 10.03.2019).

²⁾ [4] Самые популярные и удобные чат-бот сервисы. URL: <https://www.nonworkplace.com/blog/biznes/samye-populyarnye-i-udobnye-chat-bot-servisy.html>. (дата звернення 12.03.2019).

- запрошувати аудиторію на заходи, тренінги, вебінари;
- приймати і оформляти замовлення;
- збирати зворотний зв'язок і аналітичні дані про ваших клієнтів;
- здійснювати пошук важливої інформації;
- виконувати розважальну функцію.

Віртуальні співрозмовники дозволяють автоматизувати практично весь процес комунікації з аудиторією і поставити на автопілот клієнтський сервіс.

1.2 Характеристика об'єкту розробки

З огляду на відносну простоту написання програм для віртуальних помічників, легкість адаптування їх інтелекту під рішення конкретних завдань і порівняно скромні витрати, переваги чат-ботів цілком очевидні.

Для клієнта це в першу чергу істотна економія часу, підвищення ефективності пошуку і придбання товарів / послуг, а також більш звична і психологічно комфортна форма спілкування з комп'ютером.

Для підприємця ж чат-бот – це перш за все не менше істотна економія на штаті (як пряма – в рамках фонду заробітної плати, так і непряма – за рахунок скорочення робочого простору, необхідного для розміщення персоналу) [2]¹⁾.

Мета дипломного проекту – розробити повноцінного працездатного бота на Telegram, який буде надавати всю необхідну інформацію про "Школу танців", наприклад, дізнатися розклад на заданий день, адреса, телефон, види танців, вікові групи, ціна за заняття і т.д.

Такий бот стане помічником у адміністративній роботі з клієнтами школи.

¹⁾ [2] Чат-боты: электронный умный собеседник. URL: <https://robo-sapiens.ru/stati/chat-botyi/>. (дата звернення 10.03.2019).

1.3 Аналіз існуючих аналогів

Зараз багато брендів використовують чатбот для оповіщення клієнтів про акції, знижки, оновлення, для прийняття оплати і для багатьох інших цілей. Розглянемо різні види чатбот і приклади їх використання в Facebook Messenger і Telegram.

Найбільш поширені «моделі», здатні записати на прийом (до лікаря, представнику адміністрації, юристу), провести платіж, зарезервувати столик у ресторані, забронювати місце в готелі, придбати квиток (транспортний, театральний і ін.). Причому такий софт з відносно примітивним інтерфейсом зазвичай здатний написати програміст навіть із середнім рівнем кваліфікації.

Не рідкість використання чат-ботів в якості брокерів або вузькопрофільних фахівців-консультантів в таких областях, як житлово-комунальні послуги, майнові та трудові спори. Потрібні чат-боти і просто для спілкування, релаксу, психологічної підтримки [2]¹⁾.

Спочатку необхідно розглянути декілька чат-ботів для подальшого формування мети і постановки завдання до дипломного проекту.

Першим прикладом проаналізуємо роботу чатботу для продажу – «чатбот в Facebook Messenger для пошукача авіаквитків Skyscanner» (рис. 1).

Щоб купити квиток, потрібно всього лише ввести дати подорожі та місто вильоту, і сервіс покаже найдорожчі напрямки.

Чатбот в Facebook Messenger є у інтернет-магазину Ebay. У листуванні він відразу пропонує приступити до вибору товару (рис. 2). Спочатку потрібно вказати його тип (наприклад одяг для жінок), потім – бажаний стиль одягу (базовий гардероб і ін.), розмір, колір, вартість і т.д. Чатбот підбирає самі відповідні варіанти.

¹⁾ [2] Чат-боты: электронный умный собеседник. URL: <https://robo-sapiens.ru/stati/chat-botyi/>. (дата звернення 10.03.2019).

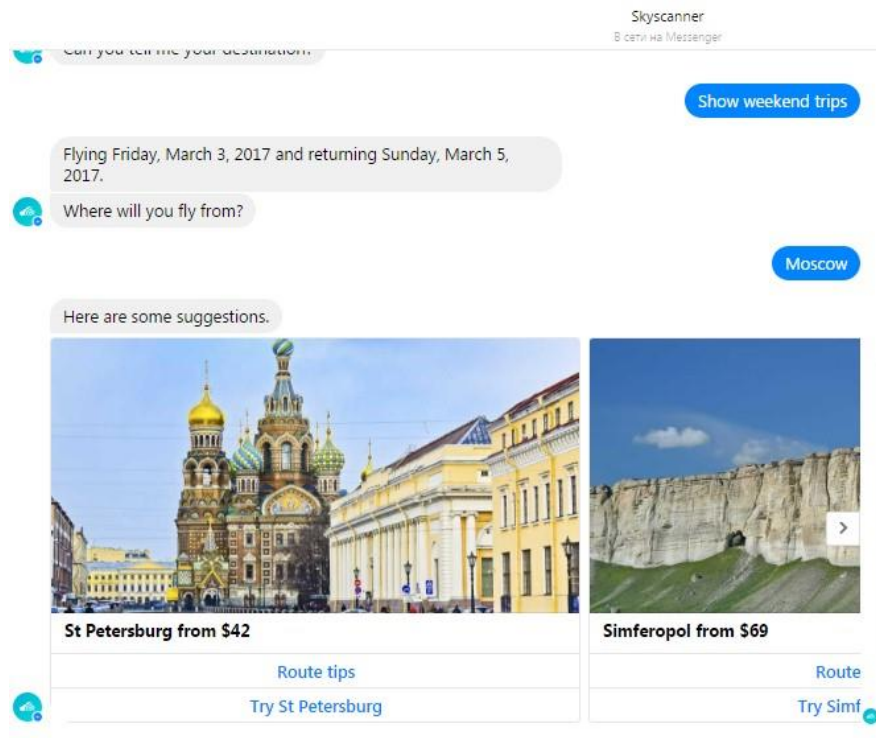


Рисунок 1 – Скрін роботи чатботу-пошукача авіаквитків Skyscanner

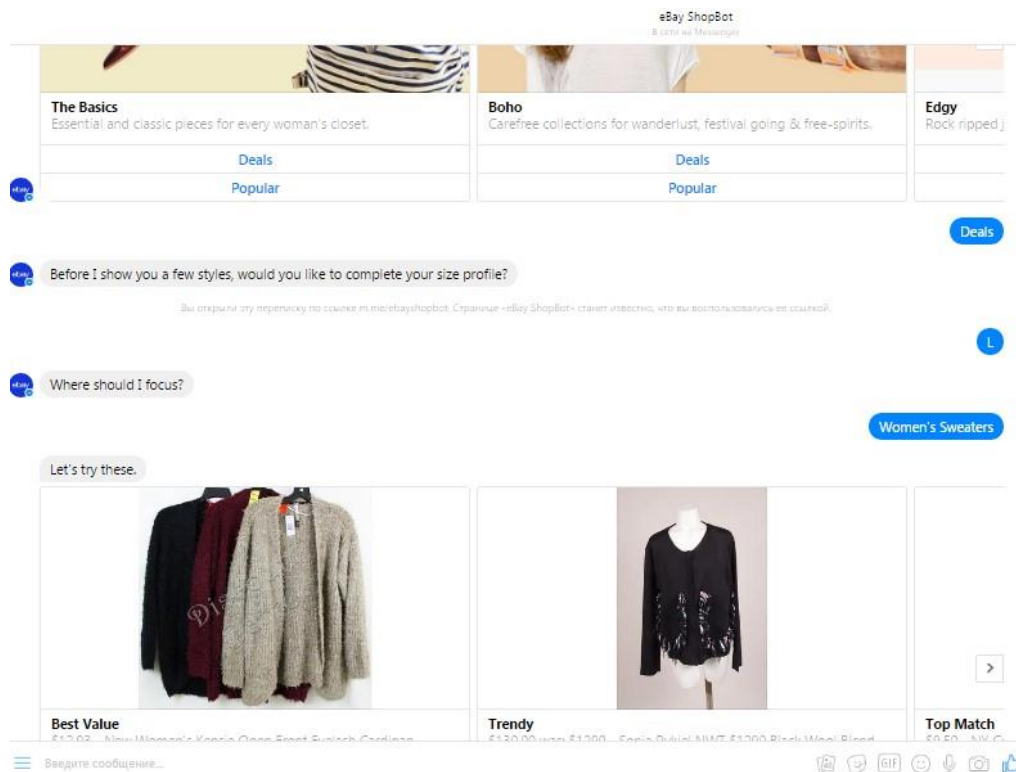


Рисунок 2 – Скрін роботи чатботу Ebay

Розглянемо ще один приклад чатботу з техпідтримки – чатбот служби підтримки магазину Sephora в Facebook Messenger (рис. 3).

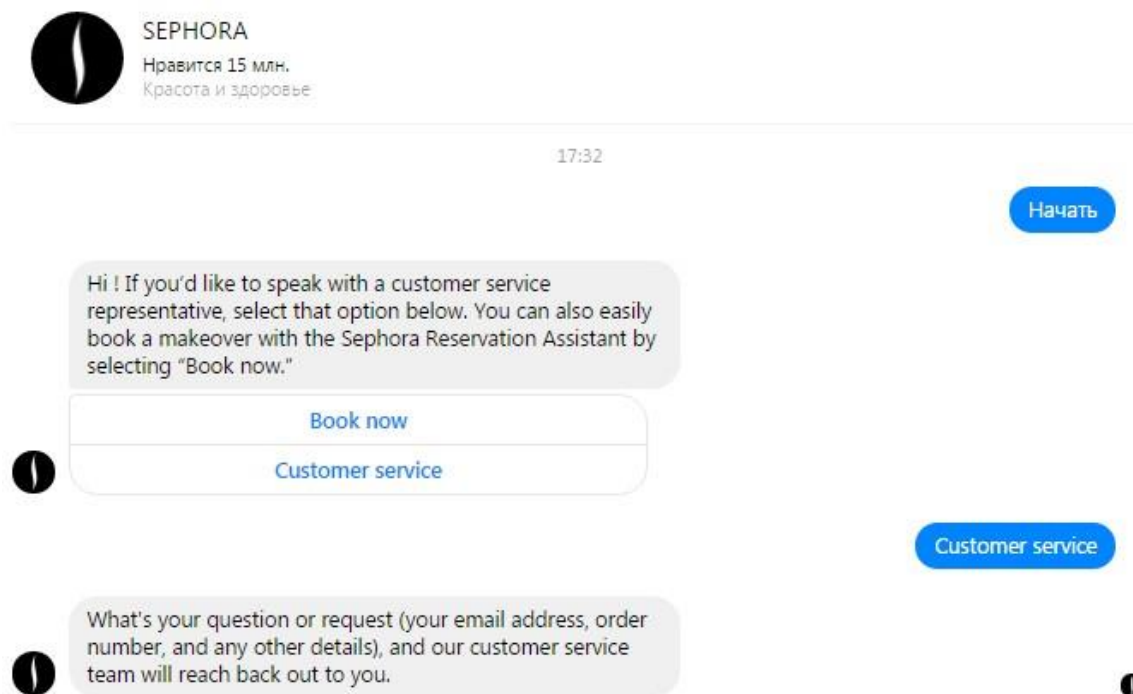


Рисунок 3 – Приклад чатботу служби підтримки магазину Sephora в Facebook

Використання чатботу для служби підтримки – повсюдно впроваджувана практика. Автоматизація процесів спілкування з клієнтами допомагає заощадити безліч часу і сил «живих» працівників компанії і виділити їм більше часу на вирішення складних завдань.

Але не всім клієнтам подобається, коли з ними спілкується бот, тому до його створення потрібно підходити дуже уважно.

Щоб чатбот видавав клієнтам точні і адекватні відповіді, потрібно внести в його пам'ять як можна більше варіацій питань і відповідей на них.

Для цього можна використовувати розділ FAQ на сайті, а також кмітливість і уяву. Приклад CNN для Facebook Messenger: до його можливостей можна віднести:

- розсилка добірок новин: останні новини, екстрені випуски, вибір редакції, добірки за темами (політика, стиль, технології і т.п.);

- пошук новин за ключовим словом і темам (рис. 4);
- показ добірки статей Top Stories за запитом.

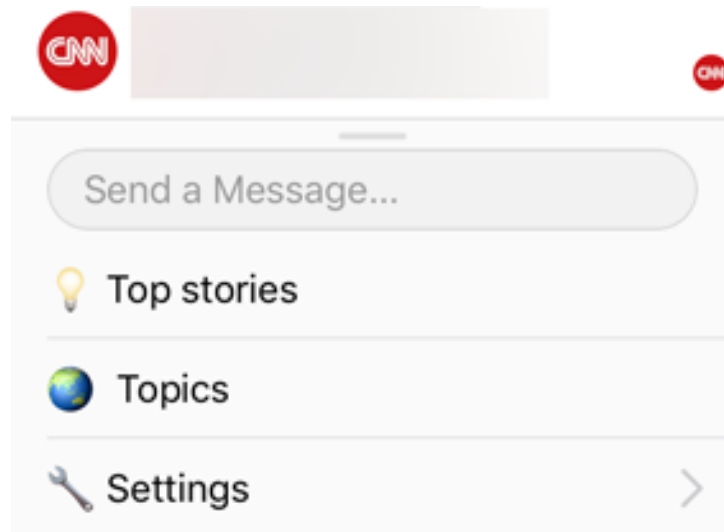


Рисунок 4 – Скрін роботи чатботу CNN: постійне меню

Добірка новин представлена у вигляді слайдера, в якому кожна картка містить ілюстрацію, заголовок, підзаголовок статті і кнопки.

При цьому, в кожній картці Top Stories всього одна кнопка, яка дозволяє прочитати статтю на сайті CNN (на мобільному пристрої – відкриває в вебв'ю, не виходячи з месенджера в мобільний браузер) [5]¹⁾.

У розсилках додається друга кнопка: «Summary», яка надсилає короткий опис статті (буквально пару слів) в текстовому повідомленні (рис. 5).

У слайдері з результатами пошуку за ключовим словом теж по дві кнопки на картку: «Читати» і «Показати подібне» (рис. 6).

Алгоритм підбору «Подібних» новин для нас поки що представляє загадку.

¹⁾ [5] Три способа запустить новостного чатбота. URL: <https://medium.com/publybot-blog>. (дата звернення 15.03.2019).

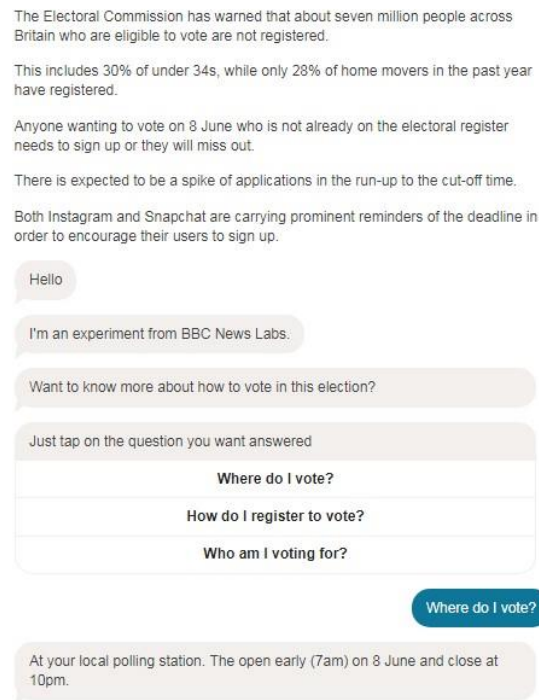


Рисунок 5 – Приклад діалогу CNN боту

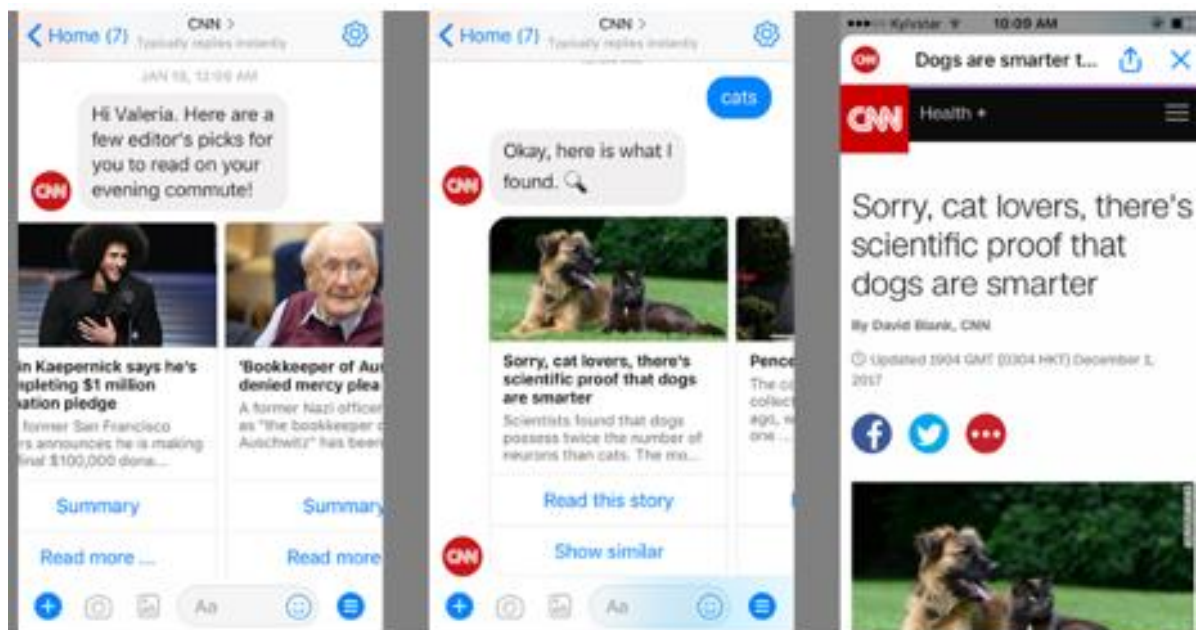


Рисунок 6 – Приклад CNN боту: три види добірок новин в слайдері і стаття

Останнім аналогом можна розглянути чатбот «HelloJarvis» (рис. 7).

«HelloJarvis» – персональний асистент, завчасно нагадує про заплановані події.



Рисунок 7 – Приклад початку роботи боту «HelloJarvis» у Facebook

Це гарний планувальник завдань, з яким можна спілкуватися людською мовою. Після нескладних налаштувань зможе обробляти фрази "завтра вранці в 7 погодувати кота", "вимкнути чайник через 4 хвилини" і багато інших (рис. 8).

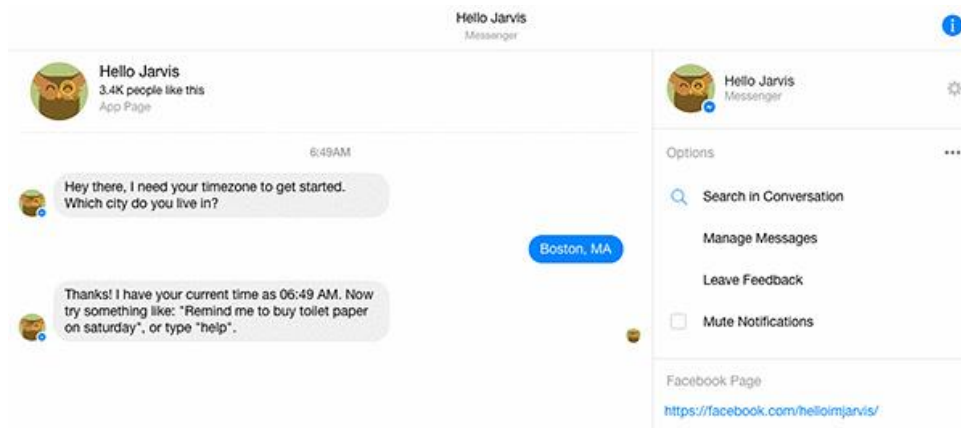


Рисунок 8 – Приклад діалогу з ботом «HelloJarvis»

Telegram-боти при вмілому використанні стають відмінною підмогою і помічниками як в повсякденних завданнях, так і в професійних потребах.

1.4 Вибір програмних засобів розробки

1.4.1 Мови програмування C# і Java

Головною особливістю мови C# є його орієнтованість на платформу Microsoft .NET – творці C# ставили собі за мету надання розробникам природних засобів доступу до всіх можливостей платформи .NET.

C# є об'єктно-орієнтованою мовою, але підтримує також і компонентно-орієнтоване програмування. Важлива особливість таких компонентів – це модель програмування на основі властивостей, методів і подій. Кожен компонент має атрибути, які надають декларативні відомості про компоненті, а також вбудовані елементи документації [6]¹⁾.

C# надає мовні конструкції, безпосередньо підтримують таку концепцію роботи. Завдяки цьому C# відмінно підходить для створення і застосування програмних компонентів.

Ось лише кілька функцій мови C#, що забезпечують надійність і стійкість додатків: прибирання сміття автоматично звільняє пам'ять, зайняту знищеними і невикористовуваними об'єктами; обробка виключень дає структурований і розширюваний спосіб виявляти і обробляти помилки; суворі типізація мови не дозволяє звертатися до не ініціалізованих змінним, виходити за межі масиву або виконувати неконтрольоване приведення типів. Щоб забезпечити сумісність програм і бібліотек C# при подальшому розвитку, при розробці C# багато уваги було приділено управлінню версіями [7]²⁾.

Таким чином, C# є нову мову програмування, орієнтований на розробку для платформи .NET і придатний як для швидкого прототипування додатків, так і для розробки великомасштабних додатків. Багато мови програмування обходять увагою це питання, і в результаті програми на цих

¹⁾ [6] Язык C#. Алексей Федоров. URL: <https://compress.ru/article.aspx?id=11719>. (дата звернення 20.03.2019).

²⁾ [7] C#. Объектно-ориентированный язык программирования. URL: <https://habr.com/ru/hub/csharp/>. (дата звернення 25.03.2019).

мовах ламаються частіше, ніж хотілося б, при виході нових версій залежних бібліотек.

Java – об'єктно-орієнтована мова програмування для написання аплетів, додатків і серверного програмного забезпечення [7]¹⁾.

Програми на Java можуть бути трансльовані в байт-код, що виконується на віртуальній java-машині (JVM) – програмою, обробній байт-код і передавальній інструкції обладнанню, як інтерпретатор, але з тією відмінністю, що байт-код, на відміну від тексту, обробляється значно швидше.

Мова Java зародився як частина проекту створення передового програмного забезпечення для різних побутових приладів. Реалізація проекту була почата на мові C++, але незабаром виник ряд проблем, найкращим засобом боротьби з якими була зміна самого інструмента – мови програмування.

Стало очевидним, що необхідний платформо-незалежний мова програмування, що дозволяє створювати програми, які не доводилося б компілювати окремо для кожної архітектури і можна було б використовувати на різних процесорах під різними операційними системами [6]²⁾.

Три ключові елементи об'єдналися в технології мови Java:

- Java надає для широкого використання свої аплети (applets) – невеликі, надійні, динамічні, які не залежать від платформи активні мережеві додатки, що вбудовуються в сторінки Web. Аплети Java можуть налаштовуватися і поширюватися споживачам з такою ж легкістю, як будь-які документи HTML;
- Java вивільняє міць об'єктно-орієнтованої розробки додатків, поєднуючи простий і знайомий синтаксис з надійним і зручним в

¹⁾ [7] C#. Объектно-ориентированный язык программирования. URL: <https://habr.com/ru/hub/csharp/>. (дата звернення 25.03.2019).

²⁾ [6] Язык C#. Алексей Федоров. URL: <https://compress.ru/article.aspx?id=11719>. (дата звернення 20.03.2019).

роботі середовищем розробки. Це дозволяє широкому колу програмістів швидко створювати нові програми і нові аплети;

- Java надає програмісту багатий набір класів об'єктів для ясного абстрагування багатьох системних функцій, використовуваних при роботі з вікнами, мережею і для введення-виведення. Ключова риса цих класів полягає в тому, що вони забезпечують створення незалежних від використовуваної платформи абстракцій для широкого спектра системних інтерфейсів.

Переваги Java як мови програмування: об'єктно-орієнтована: в Java все є об'єктом. Доповнення може бути легко розширено, тому що він заснований на об'єктній моделі. Крім цього, це платформонезалежність: на відміну від багатьох інших мов, включаючи C і C ++, Java, коли був створений, він не компілювався в платформі конкретної машини, а в незалежній від платформи байт-коді. Цей байт код поширюється через інтернет і інтерпретується в Java Virtual Machine (JVM), на якій він в даний час працює.

Java простий: процеси вивчення та введення в мову програмування Java залишаються простими. Методи перевірки автентичності засновані на шифруванні з відкритим ключем. Компілятор генерує архітектурно-нейтральні об'єкти формату файлу, що робить скомпільований код виконуваним на багатьох процесорах, з наявністю системи Java Runtime.

Java портативний: архітектурно-нейтральний і не має залежності від реалізації аспектів специфікацій – все це робить Java портативним. Компілятор в Java написаний на ANSI C з чистою переносимістю, який є підмножиною POSIX [8]¹⁾.

Java виконує зусилля, щоб усунути помилки в різних ситуаціях, спираючись в основному на час компіляції, перевірку помилок і перевірку під час виконання. Можна писати програми, які можуть виконувати безліч завдань одночасно.

¹⁾ [8] Java. URL: <http://progopedia.ru/language/java/>. (дата звернення 28.03.2019).

Введення в мову Java цієї конструктивної особливості дозволяє розробникам створювати налагоджені інтерактивні додатки.

Java байт-код переводиться на льоту в машинні інструкції та ніде не зберігається. Роблячи процес більш швидким і аналітичним, оскільки зв'язування відбувається як додаткове з невеликою вагою процесу. Введення Just-In-Time компілятора, дозволило отримати високу продуктивність.

Програмування на Java вважається більш динамічним, ніж на C або C++, так як він призначений для адаптації до мінливих умов. Програми можуть виконувати велике кількість під час обробки інформації, яка може бути використана для перевірки і дозволу доступу до об'єктів на час виконання [8]¹⁾.

1.4.2 Середа розробки Microsoft Visual Studio

Microsoft Visual Studio – це інтегроване середовище розробки з широкими можливостями для створення додатків для Windows, Android і iOS, а також сучасних веб-додатків і хмарних служб.

Професійні інструменти та служби для розробки, призначені для індивідуальних розробників або невеликих груп. Міжплатформені технології допомагають створювати додатки для Windows, Android і iOS за допомогою одного рішення для розробки.

Microsoft VS пропонує потужні функції, які допомагають швидко розібратися в програмному коді. Функція кодовою лупи допомагає зосередитися на роботі, показуючи посилання і зміни в коді, відображаючи автора останніх змін методу, а також результати проходження тестів – і все

¹⁾ [8] Java. URL: <http://progopedia.ru/language/java/>. (дата звернення 28.03.2019).

це безпосередньо в програмному коді [9]²⁾. Основні переваги IDE-середовища Visual Studio:

- вбудований Web-сервер. Для обслуговування Web-додатки ASP.NET необхідний Web-сервер, який буде очікувати Web-запити і обробляти відповідні сторінки.

Наявність в Visual Studio інтегрованого Web-сервера дозволяє запускати Web-сайт прямо з середовища проектування, а також підвищує безпеку, виключаючи ймовірність отримання доступу до тестовому Web-сайту з якого-небудь зовнішнього комп'ютера, оскільки тестовий сервер може приймати з'єднання лише з локального комп'ютера.

- підтримка безлічі мов при розробці. Visual Studio дозволяє писати код своєю рідною мовою чи будь-яких інших бажаних мовами, використовуючи весь час один і той же інтерфейс (IDE).

Більш того, Visual Studio також ще дозволяє створювати Web-сторінки на різних мовах, але поміщати їх все в один і той же Web-додаток.

- менше коду для написання. Для створення більшості додатків потрібно пристойну кількість стандартного стереотипного коду, і Web-сторінки ASP. NET тому не виключення.

Наприклад, додавання Web-елемента управління, приєднання обробників подій і коригування форматування вимагає установки в розмітці сторінки ряду деталей. У Visual Studio такі деталі встановлюються автоматично.

- інтуїтивний стиль кодування. За замовчуванням Visual Studio форматує код у міру його введення, автоматично вставляючи необхідні відступи і застосовуючи колірне кодування для виділення елементів типу коментарів.

²⁾ [9] Описание среды разработки Microsoft Visual Studio. URL: https://studbooks.net/2258619/informatika/opisanie_sredy_razrabotki_microsoft_visual_studio. (дата звернення 28.03.2019).

Такі незначні відмінності роблять код більш зручним для читання і менш схильним до помилок [9]¹⁾.

Багато з функціональних можливостей Visual Studio спрямовані на те, щоб допомагати розробнику робити свою роботу якомога швидше. Зручні функції, на зразок функції IntelliSense (яка вміє перехоплювати помилки і пропонувати правильні варіанти), функції пошуку і заміни і функції автоматичного додавання і видалення, дозволяють розробнику працювати швидко і ефективно.

- можливості налагодження. Пропоновані в Visual Studio інструменти налагодження є найкращим засобом для відстеження загадкових помилок і діагностування дивної поведінки.

Розробник може виконувати свій код по рядку за раз, встановлювати інтелектуальні точки переривання, при бажанні зберігаючи їх для використання в майбутньому, і в будь-який час переглядати поточну інформацію з пам'яті.

Visual Studio також має і безліч інших функцій:

- можливість управління проектом; вбудована функція управління вихідним кодом;
- можливість рефакторизації коду; потужна модель розширюваності.

Більш того, в разі використання Visual Studio 2008 Team System розробник отримує розширені можливості для модульного тестування, спільної роботи і управління версіями коду (що значно більше того, що пропонується в більш простих інструментах на Visual SourceSafe).

1.4.3 Середовище розробки для Java IntelliJ IDEA

¹⁾ [9] Описание среды разработки Microsoft Visual Studio. URL: https://studbooks.net/2258619/informatika/opisanie_sredy_razrabotki_microsoft_visual_studio. (дата звернення 28.03.2019).

IntelliJ IDEA середовище розробки, що було обрано для розробки дипломного проекту. IntelliJ IDEA середовище розробки для Java, що включає підтримку всіх останніх технологій і фреймворків.

Переваг у цієї програми доволі багато, але важливо те, що саме це середовище, незважаючи на те, що воно здатне підтримати багато мов програмування, зосереджено саме під розробку на мові Java. IntelliJ IDEA (рис. 9) надає інструменти для продуктивної роботи і ідеально підходить для створення комерційних, мобільних і веб-додатків [10]¹⁾.

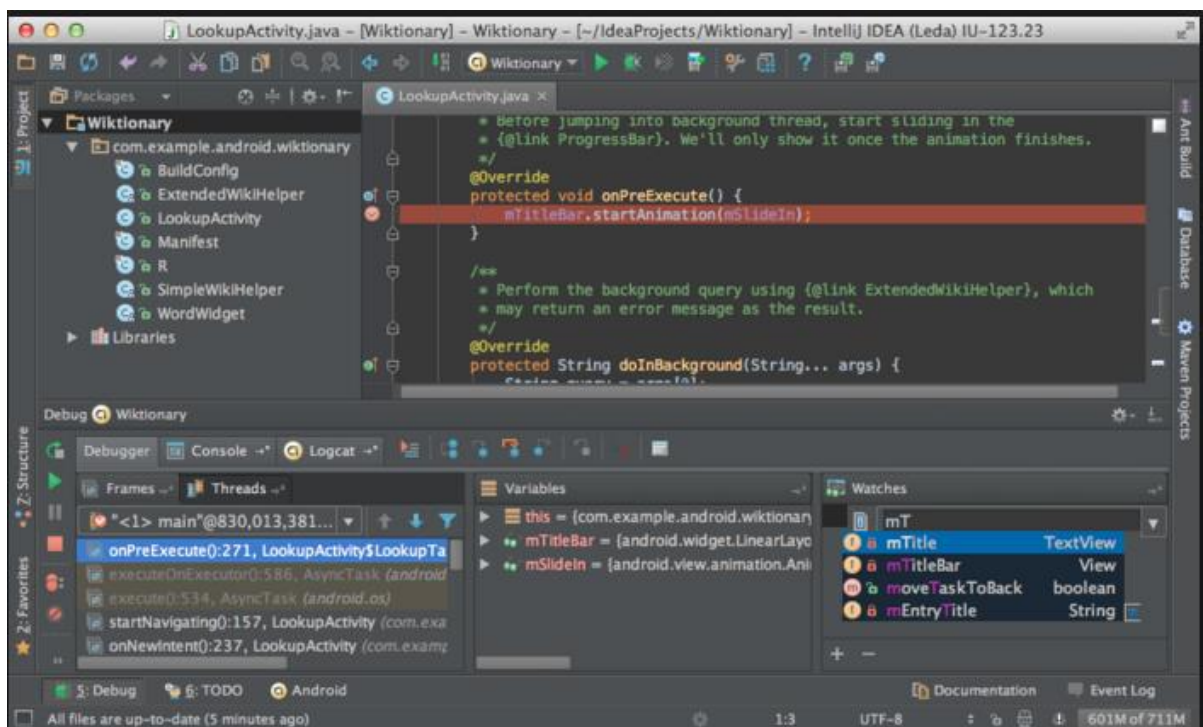


Рисунок 9 – Вікно середи розробки IntelliJ IDEA

До повної версії середи розробки включено:

- розумне автодоповнення, інструменти для аналізу якості коду, зручна навігація, розширені рефакторинг і форматування для Java, HTML, CSS, JavaScript, XML і багатьох інших мов;

¹⁾ [10] Установка и настройка IntelliJ IDEA. URL: https://geekbrains.ru/posts/intellij_idea_setup. (дата звернення 02.04.2019).

- підтримка всіх популярних фреймворків і платформ, включаючи Java EE, Spring Framework, Grails, Play Framework, GWT, AngularJS і багатьох інших;
- інтеграція з серверами додатків, включаючи Tomcat, TomEE і т.д.;
- інструменти для роботи з базами даних і SQL файлами, включаючи зручний клієнт і редактор для схеми бази даних;
- інтеграція з комерційними системами управління версіями Perforce, Team Foundation Server, ClearCase, Visual SourceSafe;
- інструменти для запуску тестів і аналізу покриття коду, включаючи підтримку всіх популярних фреймворків для тестування.

1.4.4 Інструментарій JavaFX

JavaFX являє інструментарій для створення кроссплатформенних графічних додатків на платформі Java.

JavaFX розширює можливості Java, дозволяючи розробникам використовувати будь-які бібліотеки Java в додатках JavaFX. Таким чином розробники отримують доступ до розширених можливостей Java і можуть використовувати технологію презентацій JavaFX, призначену для створення привабливих візуальних образів.

Користувачі зможуть запускати додатки JavaFX в браузері або перетягувати їх на робочий стіл.

Характеристики JavaFX:

- дозволяє розробникам інтегрувати веб-ресурси векторної графіки, анімації, аудіо і відео в багатофункціональні, інтерактивні і реалістичні додатки;
- розширює можливості технології Java, дозволяючи використовувати будь-яку бібліотеку Java в додатку JavaFX;
- дозволяє організувати ефективний робочий процес на всіх етапах від проектування до розробки. При цьому проектувальники можуть

самостійно вибрати інструменти для взаємодії з розробниками [11]¹⁾.

1.4.5 Технологія .Net Core

.NET Core представляє технологію від компанії Microsoft, призначену для створення різного роду веб-додатків: від невеликих веб-сайтів до великих веб-порталів і веб-сервісів.

.NET Core є горизонтальним розвитком програмної платформи .NET в інші операційні системи, в яких вона забезпечує можливість використання додатків, розроблених для Windows.

Одна з істотних причин головного болю програмістів – непередбачуваність умов, в яких будуть запускатися написані ними додатки. (різна конфігурація апаратної частини, різні версії операційних систем, набори бібліотек, пристрої, драйвери і т.п.).

Потужною і досить успішною спробою уніфікації програмного середовища в операційній системі Windows є платформа .NET, розроблена компанією Microsoft на початку 2000-х років [12]¹⁾.

Основна ідея полягала в тому, щоб між операційною системою і додатком помістити програмну прошарок, яка згладжувала б особливості конкретного екземпляра операційної системи, забезпечуючи додатком «ідеальну» середу виконання (рис. 10):

¹⁾ [11] Общая информация о JavaFX. URL: <https://www.java.com/ru/download/faq/javafx.xml>. (дата звернення 04.04.2019).

¹⁾ [11] Несколько слов о .NET. URL: <https://1cloud.ru/blog/chto-takoe-net-core>. (дата звернення 10.04.2019).



Рисунок 10 – Спрощено пристрій .NET

1.4.6 Система управління базами даних MySQL

MySQL – це одна з найпопулярніших і найпоширеніших СУБД (система управління базами даних) в інтернеті. Вона не призначена для роботи з великими обсягами інформації, але її застосування ідеально для інтернет сайтів, як невеликих, так і досить великих.

MySQL відрізняється хорошою швидкістю роботи, надійністю, гнучкістю. Робота з нею, як правило, не викликає великих труднощів. Підтримка сервера MySQL автоматично включається в поставку РНР.

Важливим фактором є її безкоштовність. Раніше для довготривалого зберігання інформації ми працювали з файлами: поміщали в них деяку кількість рядків, а потім витягували їх для подальшої роботи.

Завдання тривалого зберігання інформації дуже часто зустрічається в програмуванні Web-додатків: підрахунок відвідувачів в лічильнику, зберігання повідомлень у форумі, віддалене управління змістом інформації на сайті і т.д. Тим часом, професійні прийоми роботи з файлами дуже трудомісткі: необхідно піклується про приміщенні в них інформації, про її сортування, добування, при цьому не потрібно забувати, що всі ці дії будуть відбуватися на сервері хост-провайдера, де з дуже великою ймовірністю варто один з варіантів Unix.

Отже, потрібно так само піклується про права доступу до файлів і їх розміщенні. При цьому обсяг коду значно зростає, і зробити помилку в

програмі дуже просто [12]¹⁾. Всі ці проблеми вирішує використання бази даних. Бази даних самі дбають про безпеку інформації та її сортування і дозволяють отримувати і розміщувати інформацію за допомогою одного рядка.

Код з використанням бази даних виходить більш компактним, і налагоджувати його набагато легше. Крім того, не потрібно забувати і про швидкість – вибірка інформації з бази даних відбувається значно швидше, ніж з файлів.

2 ПРОЕКТНА ЧАСТИНА

2.1 Проектування інтерфейсу чатбота

2.1.1 Правила Юзабіліті для інтерфейсів

У 1990 році Якоб Нільсен сформулював 10 евристик юзабіліті для оцінки користувацьких інтерфейсів. Ці правила пройшли перевірку часом, і тепер у дизайнерів є простий спосіб оцінити, наскільки зручний інтерфейс програмного забезпечення, з опорою на універсальні принципи дизайну.

Десять евристик Нільсена:

- поінформованість про стан системи: користувачі завжди повинні знати, що на даний момент відбувається з системою. Для цього необхідна повноцінна і швидка зворотний зв'язок;
- схожість системи з реальним світом: система повинна спілкуватися з користувачем на зрозумілій йому мові. Краще використовувати слова, фрази і поняття, з якими користувач знаком, а не вузькі терміни.

Все повинно відбуватися, як в реальному світі, де інформація з'являється природно і логічно. Крім цього потрібно враховувати наступне:

¹⁾ [12] Что такое MySQL. Взаимодействие с PHP. URL: http://www.softtime.ru/bookphp/gl12_1.php. (дата звернення 12.04.2019).

- свобода дій і контроль: користувачі часто вибирають деякі функції системи помилково, і їм необхідна кнопка «аварійного виходу», щоб повернути попередній стан системи, не вступаючи в тривалий діалог.
- единбургі і стандарти: користувач не повинен сидіти і ворожити, чи означають різні слова, ситуації і дії різний або одне і те ж. Слід дотримуватися загальноприйнятих для платформи правил [13]¹⁾.
- запобігання помилок: краще хороших повідомлень про помилки може бути тільки продуманий дизайн, який допоможе запобігти проблему взагалі;
- на увазі, а не по пам'яті: слід не завантажувати пам'ять користувача, нехай всі об'єкти, дії і опції будуть на увазі. Переміщаючись по системі, користувач не повинен запам'ятовувати інформацію – все інструкції до системи повинні бути видні або легко витягатися при необхідності;
- гнучкість і ефективність: функції, що прискорюють взаємодія, яких не помітить новачок, можуть відчутно допомогти досвідченому користувачеві, і потрібно, щоб система була зручною для користувачів будь-якого рівня;
- естетика і мінімалізм в дизайні: у діалогах не повинно бути зайвої інформації. Все зайве в діалозі буде конкурувати з потрібною інформацією, роблячи її менш помітною для користувача;
- розуміння проблем та їх вирішення. Повідомлення про помилки повинні бути написані простою мовою (ніяких кодів), чітко вказувати на проблему і пропонувати конструктивне рішення;
- довідкові матеріали і документи: якби система працювала без документації, але іноді в ній з'являється необхідність.

¹⁾ [13] Usability Heuristics for Bots. URL: <https://chatbotsmagazine.com/usability-heuristics-for-bots-7075132d2c92>. (дата звернення 14.04.2019).

У користувача не повинно виникати проблем при пошуку будь-корисної інформації. Вона повинна відповідати завданням користувача, вказувати, що конкретно потрібно зробити, і не бути дуже об'ємною.

2.1.2 Проектування макетів для форм

Одним із етапів проектування є створення макетів для інтерфейсу користувача. Згідно із завданням, макет для головної форми повинен включати в себе елемент «календар» і вікно для виводу відповідної інформації щодо обраної дати (рис. 11).

Макет для вікна «Контакти» представлено на рис. 12:

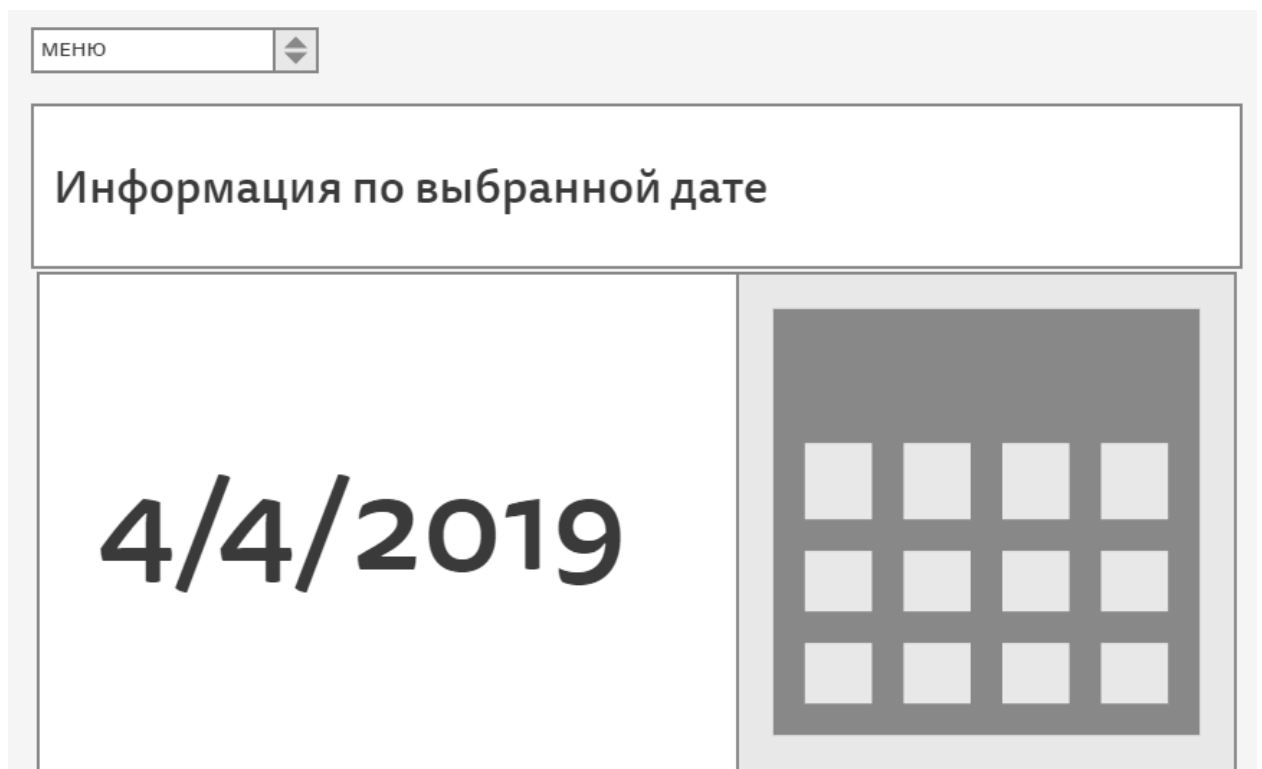
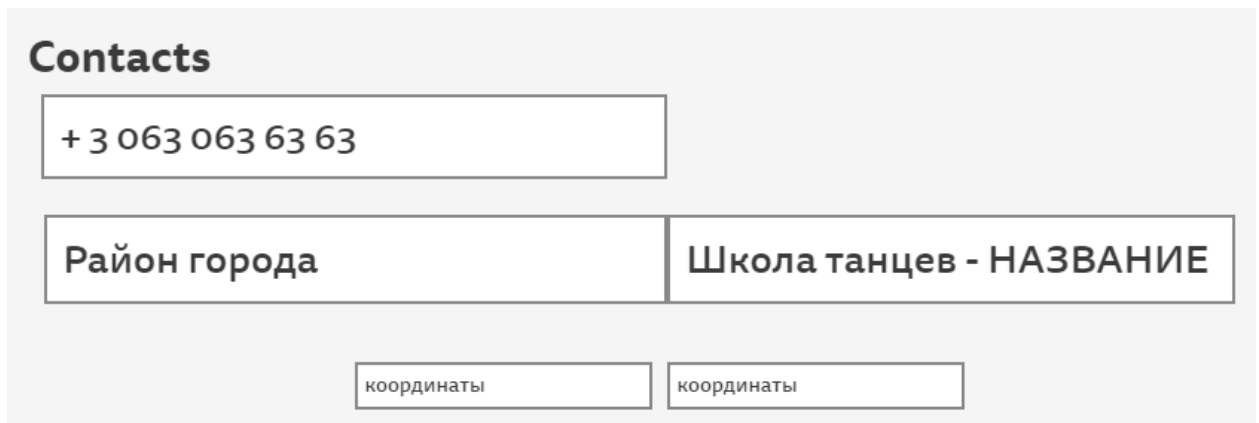


Рисунок 11 – Макет головного вікна чатботу



Contacts

+ 3 063 063 63 63

Район города Школа танцев - НАЗВАНИЕ

координаты координаты

Рисунок 12 – Макет для вікна «Контакти»

Для вікна «Види танців» спроектовано наступний макет (рис. 13).
 Макет для вікна «Возрастные группы» ,буде мати аналогічний інтерфейс.
 Проект вікна «Расписание дня» представлено на рис. 14.



Dances

Option 1

Option 2

Option 3

Name of new dance **ADD** **CLOSE**

Рисунок 13 – Макет для вікна «Види танців»

Day schedule 20.03.2019

dance	Age Group	Price	Time
Option 1			
Option 2			
Option 3			

ADD CLOSE

Рисунок 14 – Макет для вікна «Расписание дня»

Також, для оптимальної роботи боту необхідно ураховувати такий аспект, як поява нового тренування.

Для цього також буде використовуватись окрема форма, макет якої представлено на рис. 15.

New Day Record

Street dance Школьная группа

☐ FREE
☒ Selected radio button
☐ PRICE

150 uan

17:00

DONE SET TIME CLOSE

Рисунок 15 – Макет для вікна для додавання нового тренування

2.2 Алгоритми роботи чатбота

2.2.1 Алгоритм Дамерау — Левенштейн

Алгоритми нечіткого пошуку (також відомого як пошук за подібністю або fuzzy string search) є основою систем перевірки орфографії і повноцінних пошукових систем на кшталт Google або Yandex.

Наприклад, такі алгоритми використовуються для функцій на зразок «Можливо ви мали на увазі ...» в тих же пошукових системах [14]¹⁾.

Визначити «схожість» двох рядків – мінімальна кількість операцій вставки одного символу, видалення одного символу і заміни одного символу на інший, необхідних для перетворення одного рядка в іншу.

Нечіткий пошук є вкрай корисною функцією будь-якої пошукової системи. Разом з тим, його ефективна реалізація набагато складніше, ніж реалізація простого пошуку за точним збігом.

Завдання нечіткого пошуку можна сформулювати так: «По заданому слову знайти в тексті або словнику розміру n все слова, що збігаються з цим словом (або починаються з цього слова) з урахуванням k можливих відмінностей».

Наприклад, при запиті «Машина» з урахуванням двох можливих помилок, знайти слова «Машинка», «Махіна», «Малина», «Калина» і так далі [14]¹⁾.

¹⁾ [14] Нечёткий поиск в тексте и словаре. URL: <https://habr.com/ru/post/114997/>. (дата звернення 18.04.2019).

¹⁾ [14] Нечёткий поиск в тексте и словаре. URL: <https://habr.com/ru/post/114997/>. (дата звернення 18.04.2019).

Алгоритми нечіткого пошуку характеризуються метрикою – функцією відстані між двома словами, що дозволяє оцінити ступінь їх подібності в даному контексті.

Відстань Дамерау-Левенштейна: ця варіація вносить у визначення відстані Левенштейна ще одне правило – транспозиція (перестановка) двох сусідніх букв також враховується як одна операція, поряд зі вставками, вилученнями і замінами.

Ще пару років тому Фредерік Дамерау міг би гарантувати, що більшість помилок при наборі тексту – як раз і є транспозиції. Тому саме ця метрика дає найкращі результати на практиці.

2.2.2 Діаграма Use-Case для контролера запитів

Діаграми варіантів використання описують взаємини і залежності між групами варіантів використання і дійових осіб, які беруть участь в процесі.

Важливо розуміти, що діаграми варіантів використання не призначені для відображення проекту і не можуть описувати внутрішній устрій системи. Діаграми варіантів використання призначені для спрощення взаємодії з майбутніми користувачами системи, з клієнтами, і особливо знадобляться для визначення необхідних характеристик системи.

Іншими словами, діаграми варіантів використання говорять про те, що система повинна робити, не вказуючи самі застосовувані методи. Варіант використання описує, з точки зору чинного особи, групу дій в системі, які призводять до конкретного результату [15]¹⁾.

Варіанти використання є описами типових взаємодій між користувачами системи і самою системою. Вони відображають зовнішній інтерфейс системи і вказують форму того, що система повинна зробити (саме що, а не як).

¹⁾ [15] UML – диаграмма вариантов использования (use case diagram). URL: <https://habr.com/ru/post/47940/>. (дата звернення 21.04.2019).

При роботі з варіантами використання важливо пам'ятати кілька простих правил:

- кожен варіант використання відноситься як мінімум до одного дійової особи;
- кожен варіант використання має ініціатора;
- кожен варіант використання призводить до відповідного результату (результату з «бізнес-значенням»).

Дійова особа є зовнішнім джерелом (не елемент системи), який взаємодіє з системою через варіант використання. Дійові особи можуть бути як реальними людьми (наприклад, користувачами системи), так і іншими комп'ютерними системами або зовнішніми подіями [16]²⁾.

Дійові особи представляють не фізичних осіб або системи, а їх ролі. Це означає, що коли людина взаємодіє з системою різними способами (припускаючи різні ролі), він відображається декількома діючими особами.

Приклад діаграми використання для відображення принципу роботи контролера запитів представлено на рис. 16:

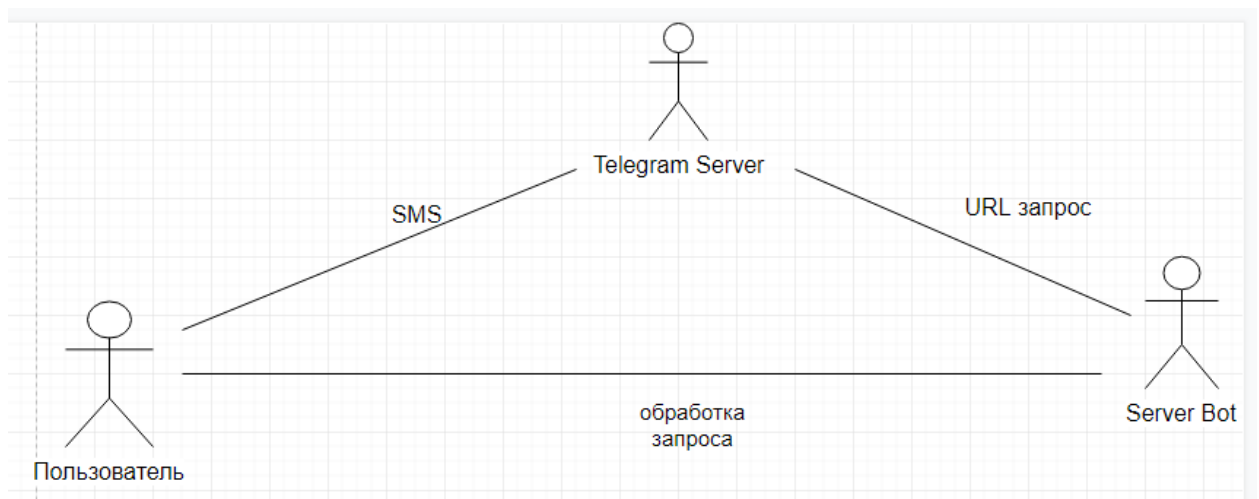


Рисунок 16 – Діаграма варіантів використання для роботи контролера запитів

²⁾ [16] Диаграмма вариантов использования (UseCase diagram). URL: https://flexberry.github.io/ru/fd_use-case-diagram.html. (дата звернення 21.04.2019).

2.2.3 Діаграма діяльності для алгоритму роботи бота

Діаграма діяльності (діаграма активності) дозволяє моделювати послідовності бізнес-процесів або дій, реалізованих методами класів. Зазначені послідовності можуть являти собою альтернативні галузей процесу обробки даних або галузям, які можуть виконуватися паралельно.

Діаграми діяльності є аналогом блок-схеми будь-якого алгоритму. Вони, як і діаграми станів та переходів, відображаються у вигляді орієнтованого графу, вершинами якого є дії, а ребрами – переходи між діями.

Діаграми діяльності доцільно використовувати для аналізу:

- змісту сценаріїв застосування проектованої системи;
- взаємодії потоків робіт різних сценаріїв;
- виконання сценаріїв у багатопроцесорних обчислювальних середовищах.

Ці діаграми широко використовуються в описі поведінки, що включає велику кількість паралельних процесів. Кожний стан на діаграмі діяльності відповідає виконанню деякої елементарної операції, а перехід в наступний стан виконується тільки після завершення цієї операції.

Таким чином, діаграму діяльності можна вважати приватним випадком діаграми станів. Основним напрямком використання діаграми діяльності є візуалізація особливостей реалізації операцій класів, коли необхідно надати алгоритми їх виконання.

Діяльність (activity) є частковим випадком стану (state) без назви, який має одну вхідну подію (OnEntry action). Тому для кожної діяльності назва складається з дієслова та декількох пояснюючих слів, наприклад «Розрахувати заробітну платню» чи «Перевірити результати запиту».

Події (events) на переходах діаграми діяльності не задаються, оскільки вважається, що перехід від однієї дії до іншої здійснюється безумовно. Гранична умова (guard condition) використовується лише для визначення дії, до якої переходить керування у випадку неоднозначності. Тобто, якщо з

даної вершини на діаграмі діяльності можна перейти до декількох інших вершин для всіх переходів необхідно визначити граничну умову. Характеристика дії (action) для переходу також не має сенсу, оскільки всі дії на цій діаграмі представлені вершинами графу.

Для діаграми діяльності характерними є наступні спеціальні стани:

- початковий стан – аналогічний до діаграми станів та переходів;
- кінцевий стан – аналогічний до діаграми станів та переходів;
- стан прийняття рішення – стан, в якому здійснюється прийняття рішення про перенаправлення потоку управління до одного зі станів, пов'язаних із даним станом;
- стан синхронізації – стан, в якому здійснюється розділення загального потоку управління на декілька гілок (чи навпаки, декілька гілок поєднуються в єдиний потік).

Діаграма діяльності для процесу оформлення тестового тренування у студії наведено на рис.17.

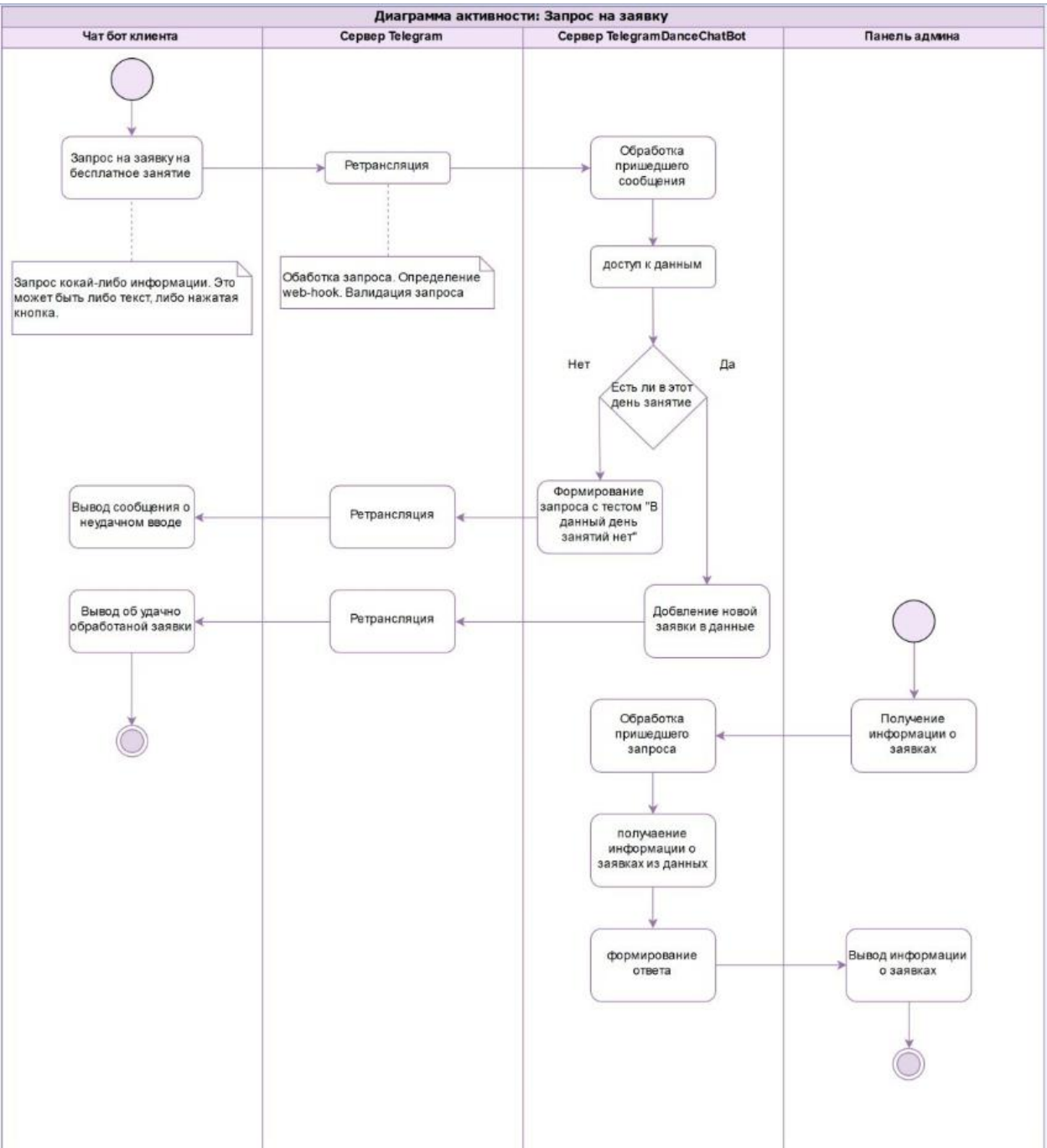


Рисунок 17 – Діаграма діяльності для процесу оформлення тестового тренування у студії

Загальна послідовність роботи боту виконує такі етапи: спочатку іде інформаційний запит від клієнта. Потім на сервері обробляються дані для реєстрації клієнта. Після обробки запиту йде доступ до інформації і формується відповідь для користувача.

Побудуємо діаграму діяльності для алгоритму роботи боту (рис. 18).



Рисунок 18 – Діаграма діяльності

2.3 Розробка структури бази даних

Для збереження інформації було прийнято рішення використовувати базу даних, структура якої наведено на рис. 19:

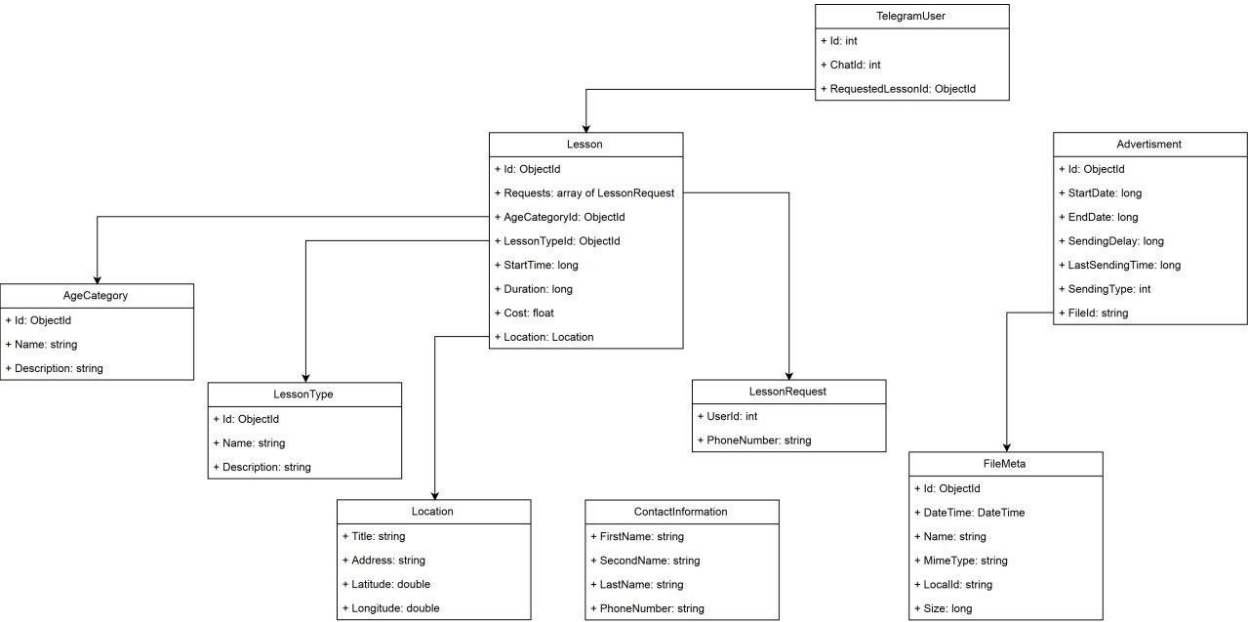


Рисунок 19 – Структура бази даних

3 РЕАЛІЗАЦІЯ ПРОЕКТУ

3.1 Сервер боту

Для реалізації ядра бота, тобто його серверної частини, була вибрана мова C#. В якості головного фреймворка був обраний DotNetCore.

3.1.1 Telegram API

Так як на C# немає готової офіційної бібліотеки для роботи з Telegram, була реалізована власна. Telegram має документацію по GET / POST запитам на сервер Telegram. Так само там описані, які сутності необхідно відправляти і приймати, і як формувати запит.

Для прикладу роботи API візьмемо відправку аудіо файлу: для початку необхідно реалізувати сутність, яка представляє собою аудіо. Спочатку створюємо відповідний клас «SendAudioModel». У його заголовку вказується анотація, яка позначає, що об'єкт цього класу повинен бути відправлений у вигляді «MultipartFormDataContent».

```
[HttpContentProperty (ContentType.MULTIPLE_PART_FORM_DATA)]
public class SendAudioModel {
```

Далі, прописуємо необхідні поля, а саме: ChatId для ідентифікації чату, куди буде відправлено повідомлення, поле disableNotification для вказівки оповіщення повідомлення, audio для безпосередньо самих даних.

```
    [JsonProperty ( "chat_id")] public object ChatId
{get; }

    [JsonProperty ( "disable_notification", NullValueHandling =
NullValueHandling.Ignore)] public bool? DisableNotification
{get; }

    [JsonProperty ( "audio")] public InputFile Audio
{get;
}
```

Над кожним властивістю присутній анотація `JsonProperty`, яка ідентифікує необхідні поля для серіалізації суті.

3.1.2 Функціонал боту та алгоритм Дамеро Левенштейн

До функціоналу боту слід віднести наступне:

- відправлення повідомлень, документів, зображень, відео, аудіо, стікер, аудіозапис, контакт, а також місця зустрічі;
- запросити адресу;
- запросити телефон;
- запросити інформацію о різновидах танців;
- запросити інформацію про вікові групи;
- запросити інформацію про розклад тренувань;
- запросити інформацію про ціну заняття.

Щоб користувачам було простіше працювати з ботом був застосований алгоритм Дамеру Левенштейн. Суть цього алгоритму в тому, щоб визначити мінімальну кількість перетворень початкового рядка в вихідний. Таким чином бот здатний правильно сприймати інформацію, введену з друкарськими помилками.

Ось частина коду, що демонструє роботу алгоритму:

```
for (int i = 1; i < sourceLength + 1; i++) {
    for (int j = 1; j < targetLength + 1; j++) {
        int cost = source.charAt(i - 1) == target.charAt(j - 1) ?
0 : 1;
        dist[i][j] = Math.min(Math.min(dist[i - 1][j] + 1,
dist[i][j - 1] + 1),
dist[i - 1][j - 1] + cost);
    }
}
```

3.1.3 Контролер запитів

Щоб сервер бота міг отримувати інформацію про повідомлення і налаштування необхідно реалізувати контролер запитів. Контролер запитів –

це клас, який має набір методів, кожен з яких налаштований під відповідний URL-запит. Якщо в чаті середовища Telegram один з користувачів щось напише, TelegramServer відправить URL-запит на сервер бота.

Бот обробляє повідомлення і відправляє відповідь за допомогою вищеписаної бібліотеки. Крім повідомлень з чату, запити можуть прийти і від адміністратора.

Нижче слід привести код, який показує принцип роботи контролера:

- створюємо клас і успадковуємо його від Controller, так вимагає фреймворк.

Зверху класу вказуємо анотацію Route, в якості вхідного параметра вказуємо доповнення до URL сервера. Це робиться для орієнтування контролера під певні запити:

```
[Route ( "api / Converting")]
public class ConvertingController: Controller
```

- далі створюється метод для отправки відео файлу. Для того, щоб фреймворк зрозумів, що метод орієнтований по POST запит, над методом ставиться анотація HttpPost.

Далі анотація Route вказує по якому запиту викликається метод:

```
[HttpPost] [Route ( "sendVideo")]
public async Task <ActionResult> SendVideo
(IFormFileCollection files, string options)
```

3.2 Опис інтерфейсу бота

3.2.1 Вікно «Сервера»

Як інтерфейс управління ботом була вибрана мова програмування Java і її глобальний фреймворк по графічним компонентам JavaFx. Перш ніж здійснювати настройку бота треба встановити зв'язок з сервером. Для цього передбачено вікно «Server Connection».

Тут вводиться IP-адреса або DNS-ім'я сервера, а також його порт (рис. 20):

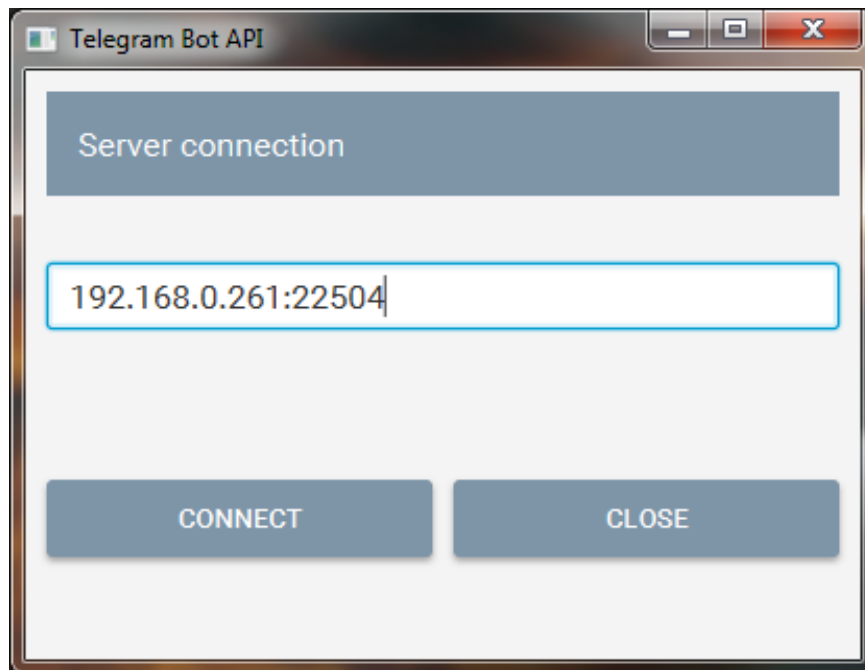


Рисунок 20— Вікно «Сервера» для бота

3.2.2 Головне меню

Після з'єднання з сервером, перед користувачем з'являється головне вікно, в якому є можливість внести різноманітні налаштування.

Так, вікно має панель «Menu» з наступними кнопками:

- «Контакти»;
- «Танці»;
- «Вікові групи»;
- «Рекламний контент».

Велику частину вікна займає календар (рис. 21), де кожен день має собою посилання на перелік подій, запланованих адміністратором школи. Натиснувши на якийсь день відкриється вікно «Розклад дня».

Дні, на які вже заплановані події, виділяються кольором.

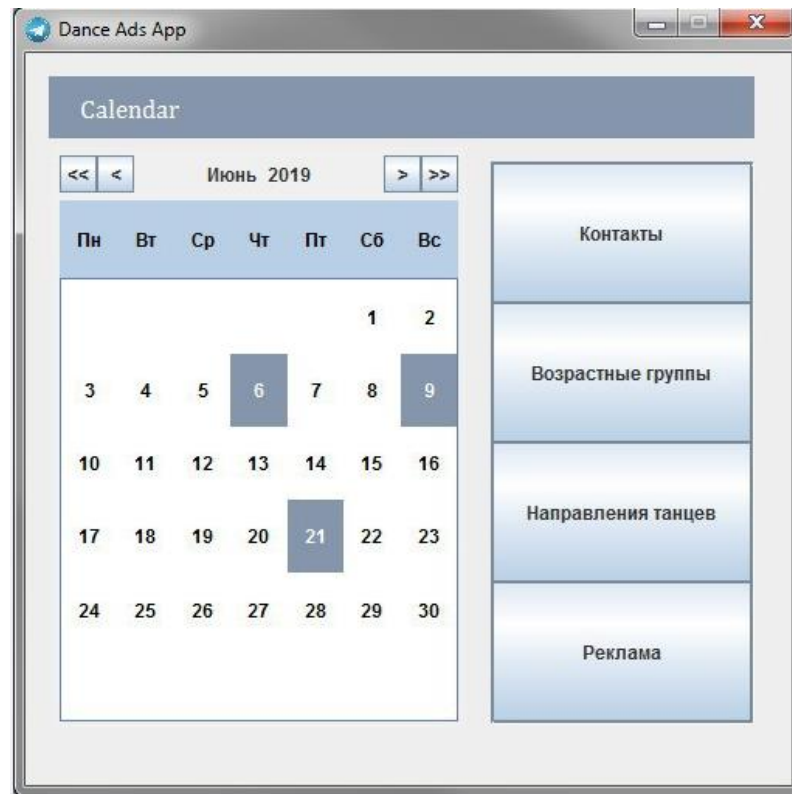


Рисунок 21 – Приклад форми «Головне вікно»

3.2.3 Вікно «Контакты»

Загальний вигляд вікна «Контакты» представлено на рис. 22.

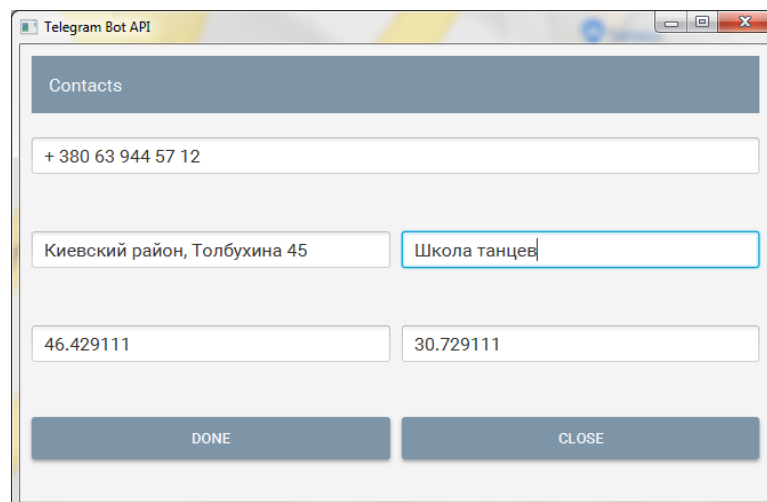


Рисунок 22 – Вікно «Контакты»

Дане вікно необхідно для контактів, які клієнт може запросити. Сюди входить номер телефону, адреса, назва школи, а також координати. Останні потрібні для того, щоб Telegram зумів відобразити шматочок карти за допомогою GoogleAPI (рис. 23).

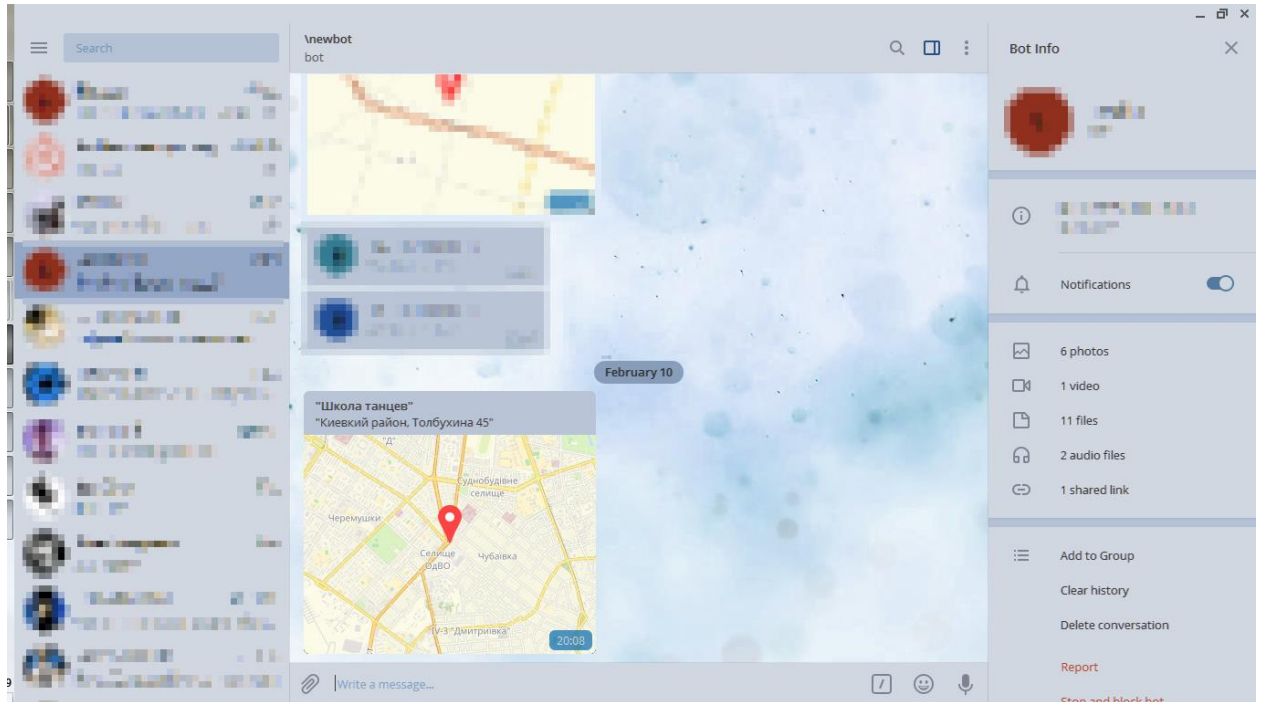


Рисунок 23 – Відображення координат

3.2.3 Вікно «Види танців»

Тут можна вказати всі види танцювального напрямку, які надає школа. Цей список можуть запитувати клієнти, а так же цей список включено до синхронізації у вікні «Новий запис» (рис. 24). Це запобігає помилки і постійний введення одного і того ж тексту.

На даний час у базі заложенні наступні записи до цієї категорії:

- самба;
- румба;
- Twerk;

- хастл;
- народні танці;
- вуличні танці;
- брейк та інше.

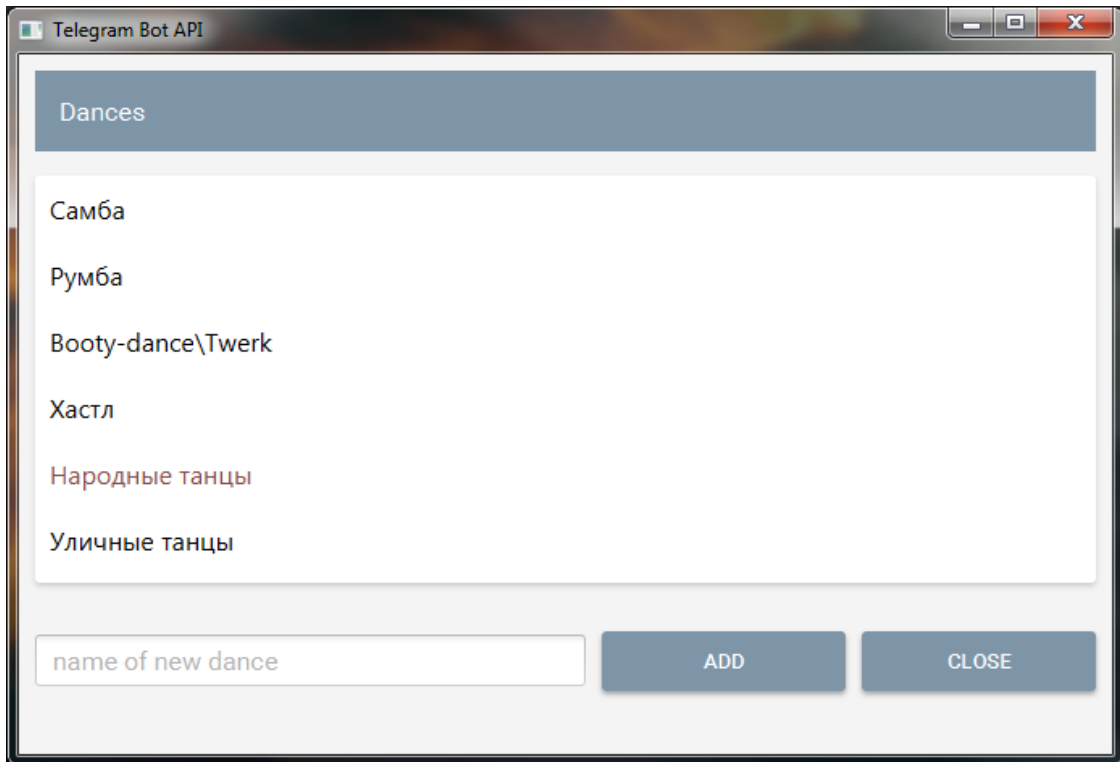


Рисунок 24 – Вікно «Види танців»

3.2.4 Вікно «Возрасные группы»

Вікно «Возрасные группы» (рис. 25): тут можна вказати всі вікові групи, які надає школа. Крім цього, це вікно потрібне для введення нового напрямку танців у разі появи останнього у школі.

Як і в попередньому вікні, цей список можуть запитувати клієнти, а також цей список використовується у вікні «Новий запис».

Вікно «Розклад дня»: тут налаштовується інформація про тренування в якийсь конкретний день. До уваги користувача надається таблиця, де вказані

напрямок танців, вікова група, ціна за заняття і час. Натиснувши на кнопку «Add», з'являється вікно, де можна вказати нове заняття (рис. 26).

Telegram Bot API

Age groups

- младшая группа (3 - 5 годиков)
- Школьная группа(6 - 12 лет)
- Подростковая группа(6 - 11 лет)
- Старшая группа(12 - 18 лет)
- Взраслая группа(от 18 лет)

name of new group

ADD CLOSE

Рисунок 25 – Вікно «Возрасные группы»

Telegram Bot API

Day schedule 20.03.19

Dance ▲	Age group	Price	Time
Народные танцы	младшая группа (3 - 5 годиков)	200грн	12:00
Самба	Подростковая группа(6 - 11 лет)	200грн	09:00
Хастл	Старшая группа(12 - 18 лет)	500грн	10:30

ADD CLOSE

Рисунок 26 – Приклад виводу інформації щодо заданої дати

3.2.5 Вікна «Новий запис»

Дане вікно розроблене для адміністратора. Воно використовується для створення нової тренування. Тут вказуються вид танців, група, ціна заняття (є можливість поставити безкоштовно) і час початку заняття (рис. 27).

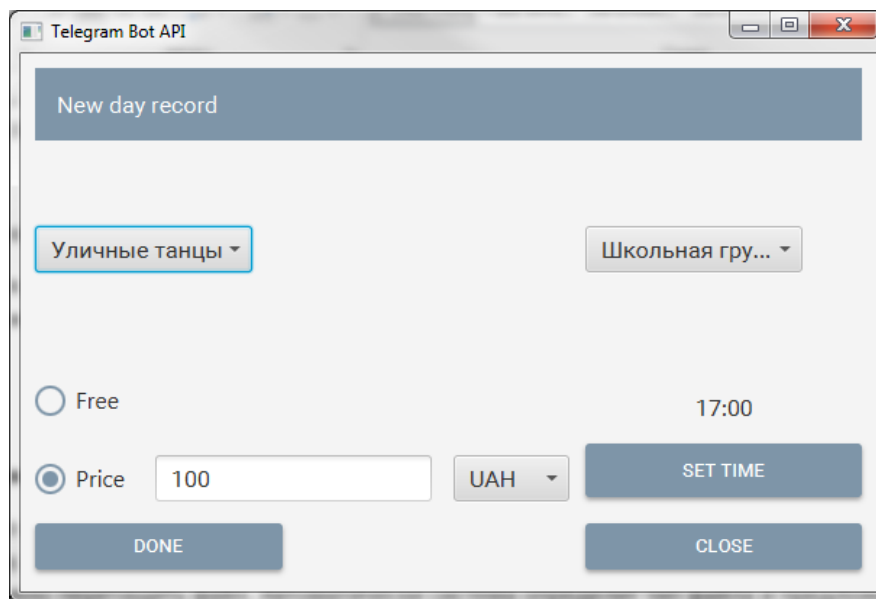


Рисунок 27 – Приклад створення нового запису

3.2.6 Вікно «Рекламний контент»

Тут можна вказати якусь рекламну інформацію, наприклад, відео виступ або рекламну обкладинку. Так само є можливість відправляти документи. У вікні є місце для «Drag&Drop», куди можна перетягнути файл (рис. 28).

Автоматично система визначить тип файлу і запропонує кілька варіантів його відправки. Справа в тому, що Telegram дозволяє відправляти файли в вигляді документа, і тоді файл приходить без змін. Але його треба завантажити і відкрити за допомогою своїх засобів, або ж відправити його як відео або картинку, і тоді можна його відкрити всередині програми Telegram.

Правда Telegram файл перекодує, знижуючи при цьому якість.

Далі вказується дата початку відправлення реклами в чат, дата закінчення і частота відправки. Мінімальна частота 10 хвилин.

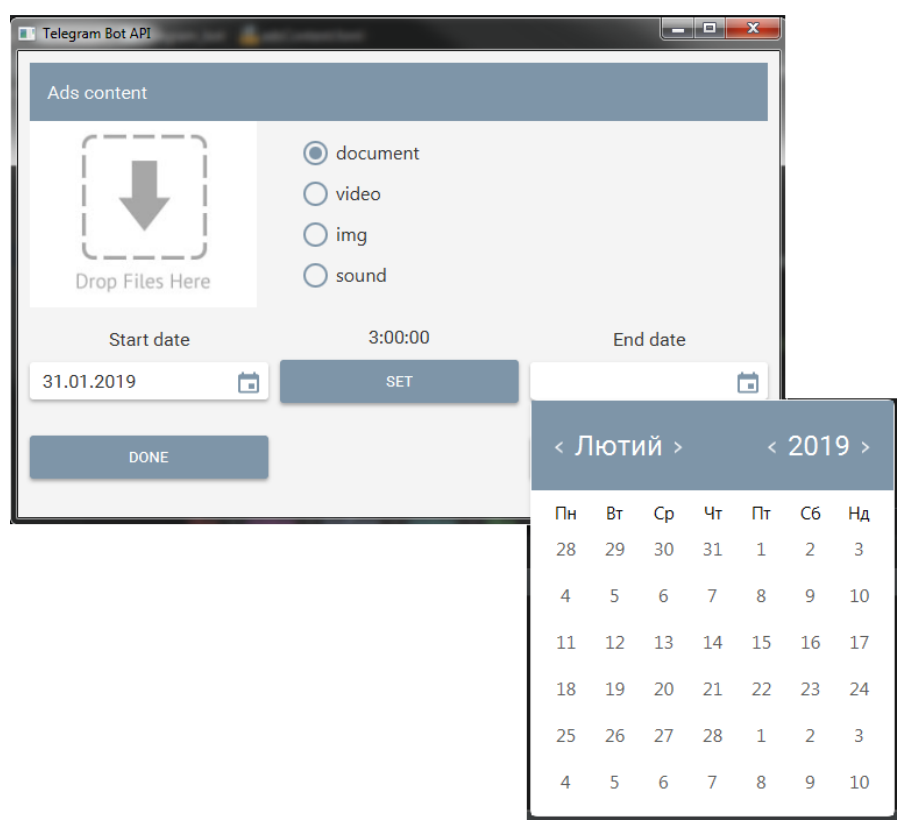


Рисунок 28 – Налаштування рекламного контенту

ВИСНОВКИ

Тема чатботів, безсумнівно, на сьогоднішній день користується популярністю. Вони можуть стати гарними помічниками-секретарями. До функцій чатбота може входити багато аспектів, таких як:

- встановлення мови для чатботу;
- автопостінг інформації з соціальних мереж і відеохостингів;
- налаштування автоматичної трансляції в канал Telegram.

Під час аналітичного огляду предметної області «чатботи» було виконано:

- аналітичний огляд матеріалу по розробки чатботу, а саме створення програмних додатків під мобільні пристрої;
- зроблено аналіз топових аналогів об'єкту розробки, на основі якого поставлені вимоги і сформовані мета роботи;
- обрані програмні засоби для реалізації проекту.

Результати стадії проектування:

- згідно з правилами юзабіліті спроектовані макети форм для роботи користувача з чатботом;
- побудована діаграма варіантів використання для роботи контролеру запитів;
- для роботи чатботу обрано алгоритм Дамерау — Левенштейн.

Етап реалізації проекту:

- створено функціонал чатбота з використанням обраних засобів розробки;
- розроблені форми для його взаємодії з користувачами.

В рамках подальшого розвитку проекту чатбота буде підключено до Telegram для тестування його роботи у режимі реального часу. Крім цього, до його головної форми буде добавлені кнопки з основними питаннями, які користувачі найчастіше задають для отримання інформації щодо школи танців, для ускорення діалогу з ботом.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Чат-боты – следующее поколение приложений. URL: <https://habr.com/ru/company/microsoft/blog/281459/>. (дата звернення 10.03.2019).
2. Чат-боты: электронный умный собеседник. URL: <https://robo-sapiens.ru/stati/chat-botyi/>. (дата звернення 10.03.2019).
3. Почему чатботы захватывают мир + 25 примеров для вдохновения. URL: <https://texterra.ru/blog/pochemu-chatboty-zakhvatyvayut-mir-25-primerov-dlya-vdokhnoveniya.html>. (дата звернення 12.03.2019).
4. Самые популярные и удобные чат-бот сервисы. URL: <https://www.nonworkplace.com/blog/biznes/samye-populyarnye-i-udobnye-chat-bot-servisy.html>. (дата звернення 12.03.2019).
5. Три способа запустить новостного чатбота. URL: <https://medium.com/publybot-blog>. (дата звернення 15.03.2019).
6. Язык C#. Алексей Федоров. URL: <https://compress.ru/article.aspx?id=11719>. (дата звернення 20.03.2019).
7. C#. Объектно-ориентированный язык программирования. URL: <https://habr.com/ru/hub/csharp/>. (дата звернення 25.03.2019).
8. Java. URL: <http://progopedia.ru/language/java/>. (дата звернення 28.03.2019).
9. Описание среды разработки Microsoft Visual Studio. URL: https://studbooks.net/2258619/informatika/opisanie_sredy_razrabotki_microsoft_visual_studio. (дата звернення 28.03.2019).
10. Установка и настройка IntelliJ IDEA. URL: https://geekbrains.ru/posts/intellij_idea_setup. (дата звернення 02.04.2019).
11. Несколько слов о .NET. URL: <https://1cloud.ru/blog/chto-takoe-net-core>. (дата звернення 10.04.2019).

12. Что такое MySQL. Взаимодействие с PHP. URL:
http://www.softtime.ru/bookphp/gl12_1.php. (дата звернения 12.04.2019).
13. Usability Heuristics for Bots. URL:
<https://chatbotsmagazine.com/usability-heuristics-for-bots-7075132d2c92>.
(дата звернения 14.04.2019).
14. Нечёткий поиск в тексте и словаре. URL:
<https://habr.com/ru/post/114997/>. (дата звернения 18.04.2019).
15. UML – диаграмма вариантов использования (use case diagram). URL:
<https://habr.com/ru/post/47940/>. (дата звернения 21.04.2019).
16. Диаграмма вариантов использования (UseCase diagram). URL:
https://flexberry.github.io/ru/fd_use-case-diagram.html. (дата звернения 21.04.2019).