**Київський національний торговельно-економічний університет**

**Кафедра комп’ютерних наук та інформаційних систем**

**ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЕКТ**

на тему:

**«Створення інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram.»**

|  |  |  |
| --- | --- | --- |
| Студента 1 курсу, 3м групи,  спеціальності  122 «Комп’ютерні науки» | \_\_\_\_\_\_\_\_\_\_  *підпис студента* | Солодкого Миколи Андрійовича |
| Науковий керівник  кандидат фізико-математичних наук, доцент | \_\_\_\_\_\_\_\_\_\_\_  *підпис керівника* | Самойленко Ганна Тимофіївна |
| Гарант освітньої програми  доктор фізико-математичних наук, професор | \_\_\_\_\_\_\_\_\_\_\_  *підпис керівника* | Пурський Олег Іванович |

**Київ 2021**

**Київський національний торговельно-економічний університет**

Факультет інформаційних технологій

Кафедра комп’ютерних наук та інформаційних систем

Спеціальність 122 «Комп’ютерні науки»

**Затверджую**

Зав. кафедри \_\_\_\_\_\_\_\_\_\_\_\_Пурський О.І.

«20» грудня 2020р.

**Завдання**

**на випускний кваліфікаційний проект студенту**

**Солодкому Миколі Андрійовичу**

*(прізвище, ім’я, по батькові)*

**1. Тема випускної кваліфікаційної роботи (проекту)**

*«Створення інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram.»*

Затверджена наказом ректора від *«02» грудня 2020 р. № 4110*

**2. Строк здачі студентом закінченої роботи** *25 листопада 2021 року*

**3. Цільова установка та вихідні дані до роботи**

*Мета роботи: обґрунтування та розробка мобільного веб-додатку, з урахуванням сучасних світових тенденцій побудови організаційних та функціональних інформаційних структур підприємств.*

*Об’єкт дослідження: процес проектування мобільного web-додатку для електроннї торгівлі*

*Предмет дослідження: засоби створення мобільного web-додатку для електроннї торгівлі*

**4. Перелік графічного матеріалу** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:**

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Консультант  (прізвище, ініціали) | Підпис, дата | |
| Завдання видав | Завдання прийняв |
| 1 | Самойленко Г.Т. | 5.12.2020р. | 5.12.2020р. |
| 2 | Самойленко Г.Т. | 5.12.2020р. | 5.12.2020р. |
| 3 | Самойленко Г.Т. | 5.12.2020р. | 5.12.2020р. |

**6. Зміст випускної кваліфікаційної роботи (проекту) (перелік питань за кожним розділом)**

*ВСТУП*

*РОЗДІЛ 1. Теоретичні основи Web-технологій*

*1.3. Основні поняття Web-технологій*

*Мови Web-програмування, їх різновиди та функції*

*Різновиди ботів та їх застосування у Web.*

*Висновки до Розділу 1*

*РОЗДІЛ 2. Організація розробки Web-додатку.*

*2.1. Особливості мови програмування Python*

*2.2. Використання мови Python для розробки Telegram бот*

*2.3 Опис існуючих Telegram ботів.*

*Висновки до Розділу 2*

*РОЗДІЛ 3. Програмна реалізація інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram*

*3.1. Розробка структури і інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram*

*3.2 Розробка моделі функціонування інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram*

*3.3. Пpoгpaмнa peaлiзaцiя проекту*

*Висновки до Розділу 3*

*ВИСНОВОК*

*СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ*

**7. Календарний план виконання роботи**

|  |  |  |  |
| --- | --- | --- | --- |
| №  Пор. | Назва етапів випускного  кваліфікаційного проекту | Строк виконання етапів роботи | |
| За планом | фактично |
| 1 | 2 | 3 | 4 |
| 1 | *Вибір теми випускної кваліфікаційної роботи* | *02.10.2020* | *02.10.2020* |
| 2 | *Розробка та затвердження завдання на випускну кваліфікаційну роботу* | *05.12.2020* | *05.12.2020* |
| 3 | *Вступ* | *01.06.2021* |  |
| 4 | *РОЗДІЛ 1. Теоретичні основи Web-технологій* | *25.06.2021* |  |
| 5 | *РОЗДІЛ 2. Організація розробки Web-додатку.* |  |  |
| 6 | *РОЗДІЛ 3. Програмна реалізація інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram* | *02.09.2021* |  |
| 7 | *Висновки* | *02.11.2021* |  |
| 8 | *Підготовка статті у збірник наукових статей магістрів* | *09.06.2021* |  |
| 9 | *Здача випускної кваліфікаційної роботи на кафедру науковому керівнику* | *05.11.2021* |  |
| 10 | *Попередній захист випускної кваліфікаційної роботи* | *20.11.2021* |  |
| 11 | *Виправлення зауважень, зовнішнє рецензування випускної кваліфікаційної роботи* | *22.11.2021* |  |
| 12 | *Представлення готової зшитої випускної кваліфікаційної роботи на кафедру* | *25.11.2021* |  |
| 13 | *Публічний захист випускної кваліфікаційної роботи* | *За розкладом роботи ЕК* |  |

**8. Дата видачі завдання** «5» грудня 2020 р.

Керівник випускного кваліфікаційного проекту

Самойленко Г.Т.

*(прізвище, ініціали, підпис)*

Гарант освітньої програми Пурський О.І.

*(прізвище, ініціали, підпис)*

Завдання прийняв студент-дипломник Солодкий М.А.

*(прізвище, ініціали, підпис)*

**9. Відгук керівника випускної кваліфікаційної роботи (проекту)**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Керівник випускного кваліфікаційного проекту

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_.\_\_\_\_\_\_\_\_ 20\_\_ р.

*(підпис, дата)*

**10. Висновок про випускний кваліфікаційний проект**

Випускний кваліфікаційний проект студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(прізвище, ініціали)*

може бути допущена до захисту в екзаменаційній комісії.

Гарант освітньої програми Пурський О.І.

*(підпис, прізвище, ініціали)*

Завідувач кафедри Пурський О.І.

*(підпис, прізвище, ініціали)*

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021 р.

**Анотація**

У випускному кваліфікаційному проекті проаналізовано поняття «електронний розклад студента», його можливості в умовах інформатизації навчального процесу, структуру та види інформаційних систем. Розглянуто види структурування даних та види представлення даних. Виконане структурування даних для зручного їх представлення у інформаційній системі. Здійснена розробка діаграм візуалізації інформаційної системи. Розроблено метод відображення структурованих даних на сторінці веб-додатку. Створено веб-додаток інформування студента щодо розкладу занять на поточний чи наступний тиждень.

**Ключові слова:** веб-додаток, діаграма візуалізації, електронний розклад, інформаційна система.

**Annotation**

The graduation qualification project analyzes the concept of "electronic student schedule", its capabilities in terms of informatization of the educational process, the structure and types of information systems. Types of data structuring and types of data representation are considered. Structuring of data for their convenient representation in information system is executed. The development of information system visualization diagrams has been carried out. A method for displaying structured data on a web application page has been developed. A web application for informing students about the schedule of classes for the current or next week has been created.

**Keywords:** web application, visualization diagram, electronic schedule, information system.

**ЗМІСТ**

[**ВСТУП** 10](#_Toc85649063)

[**РОЗДІЛ 1.**](#_Toc85649064) [**ТЕОРЕТИЧНІ ОСНОВИ WEB-ТЕХНОЛОГІЙ** 11](#_Toc85649065)

[1.1. Основні поняття Web-технологій 11](#_Toc85649066)

[1.2. Мови Web-програмування, їх різновиди та функції 13](#_Toc85649067)

[1.3. Різновиди ботів та їх застосування у Web. 16](#_Toc85649068)

[Висновки до Розділу 1 19](#_Toc85649069)

[**РОЗДІЛ 2.**](#_Toc85649070) [**ОРГАНІЗАЦІЯ РОЗРОБКИ WEB-ДОДАТКУ** 20](#_Toc85649071)

[2.1. Огляд конструкторів чат-ботів 20](#_Toc85649072)

[2.2. Особливості мови програмування Python 21](#_Toc85649073)

[2.3 Опис існуючих Telegram ботів. 23](#_Toc85649074)

[Висновки до Розділу 2 27](#_Toc85649075)

[**РОЗДІЛ 3.**](#_Toc85649076) [**ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОГО ЧАТ-БОТА ДЛЯ ПІДТРИМКИ ОСВІТНЬОГО ПРОЦЕСУ В СЕРЕДОВИЩІ TELEGRAM** 28](#_Toc85649077)

[3.1. Розробка структури інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram 28](#_Toc85649078)

[3.2. Розробка моделі функціонування інформаційного чат-боту для підтримки освітнього процесу в середовищі Telegram 29](#_Toc85649079)

[3.3. Пpoгpaмнa peaлiзaцiя проекту 32](#_Toc85649080)

[Висновки до Розділу 3 32](#_Toc85649081)

[**ВИСНОВОК** 32](#_Toc85649082)

[**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ** 33](#_Toc85649083)

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ**

WWW (англ. World Wide Web) - це розподiлена мережева гiпертекстова iнформацiйна система.

БД – база даних.

СУБД – система управління базами даних.

SQL (англ. Structured Query Language) - мова структурованих запитів.

AJAX – (англ. Asynchronous JavascriptAnd Xml), технологія звернення до сервера без перезавантаження сторінки.

CSS – (англ. Cascading Style Sheets), каскадні таблиці стилів.

HTML – (англ. HyperText Markup Language), мова розмітки гіпертексту.

API – (англ. Application Programming Interface), набір чітко визначених методів для взаємодії різних компонентів.

MVVM (англ. Model-View-ViewModel) - шаблон проєктування, що застосовується під час створення архітектури застосунків (додатків) і дозволяє відокремити логіку додатку від його візуальної частини.

MVW (англ. Model-View-Whatever) парадигма - це термін, що вказує на можливості фреймворку, що використовується, надати можливість вибору між шаблонами MVC або MVVM.

SPA (англ. Single Page Application) - веб-додаток або веб-сайт, який взаємодіє з користувачем шляхом динамічного переписування поточної сторінки.

DOM (англ. Document Object Model) - це модель документа, завантажена у браузері, що представляє документ у вигляді дерева вузлів, де кожен вузол представляє частину документа (наприклад, елемент, текстовий рядок або коментар).

IDEF0 - це методологія графічного опису систем і процесів діяльності організації як безлічі взаємозалежних функцій.

HTTP (англ. Hypertext Transfer Protocol) - протокол передачі гіпертексту.

# **ВСТУП**

# **РОЗДІЛ 1.**

# **ТЕОРЕТИЧНІ ОСНОВИ WEB-ТЕХНОЛОГІЙ**

## **Основні поняття Web-технологій**

Веб-розробка містить величезний набір правил і методів, про які повинен знати кожен розробник веб-сайтів. Для правильної роботи веб-сайтів та інтернет-сервісів необхідно знати веб-технології, які дозволяють підтримувати їх роботу.

**Веб-технологія** – комплекс технічних, комунікаційних, програмних методів розв’язання завдань організації спільної діяльності користувачів із застосування мережі Інтернет.[1] Іншими словами, веб-технології - це інфраструктурні будівельні блоки будь-якої ефективної комп’ютерної мережі.

Поняття «веб-технології» пов’язане з використанням глобального інформаційного простору, що заснований на фізичній інфраструктурі Інтернету і протоколі передачі даних HTTP або веб-простору (WWW).

Веб-технології включають наступне:

* Основи, які включають веб-браузери та деякі основи розробки веб-додатків;
* Мови програмування та фреймворки, які використовуються при розробці веб-сайтів;
* Бази даних, які використовуються на сервері для зберігання даних, необхідних або зібраних веб-сайтами;
* Деякі протоколи, тобто правила спілкування в Інтернеті;
* Графічні, аудіовізуальні та інші мультимедійні елементи;
* Деякі формати даних, які зазвичай використовуються для передачі даних через Інтернет;
* Інші різні веб-технології.[2]

Для опрацювання різних видів інформації в мережі достатньо використовувати браузер і набір відповідних веб-технологій, які функціонують в мережі на різних серверах.

**Браузер** - програма, що дозволяє показ і взаємодію з ресурсами, присутніми на сторінках веб-сайтів у всесвітній мережі Інтернет або локальній мережі.[3] Вони засновані на клієнт-серверній архітектурі. У цьому випадку клієнтом є браузер.

**Сервер** є комбінацією програмного та апаратного забезпечення, що використовує HTTP та інші протоколи, що має на меті відображати контент веб-сайту зберігаючи, оброблюючи і надсилаючи веб-сторінки клієнту. Спілкування між клієнтом і сервером відбувається в режимі «запит-відповідь». Кожен обчислювальний процес, який використовується або викликається іншим процесом (особливо віддалено), і є сервер; а процес є клієнтом.

**Веб-розробка** – це процес створення веб-сайтів або веб-додатків. Основними етапами є:

* веб-дизайн,
* верстка сторінок,
* програмування для веб на стороні клієнта і сервера,
* конфігурування веб-сервера.

**База даних (БД)** — це організована структура, яка призначена для зберігання, зміни та обробки взаємозалежної інформації, переважно великих обсягів. БД використовують для динамічних сайтів з великими обсягами (інтернет-магазин, портал, корпоративний сайт).

Головною перевагою БД є швидкість внесення та використання потрібної інформації. Завдяки спеціальним алгоритмам, які використовуються для баз даних, можна легко знаходити необхідні дані всього за декілька секунд. Також в базі даних існує певний взаємозв'язок інформації: зміна в одному рядку може спричинити зміни в інших рядках — це допомагає працювати з інформацією простіше і швидше.

Бази даних для сайтів дають змогу зберігати інформацію, що виглядає як зв'язані між собою таблиці. Саме в БД зберігаються вся необхідна та корисна інформація для функціонування сайту (клієнтські дані, прайс-лист, список товарів).

Щоб створити запит до бази даних часто використовують Structured Query Language. SQL дає змогу додавати, редагувати та видаляти інформацію, що міститься у таблицях.

Під час програмування сайтів використовують різні системи управління БД. До основних СУБД, відносять:

* система керування об'єктно-реляційними базами даних Oracle Database;
* **система керування реляційними базами даних** MySQL;
* **система керування реляційними базами даних Microsoft SQL Server;**
* **система керування реляційними базами даних PostgreSQL.[4]**

**Протокол передачі гіпертексту**, більш відомий як HTTP, - це веб-протокол, який визначає два поняття:

* Як клієнтські запити передаються на сервери
* Як сервери реагують на запити клієнтів

Графічними елементами є:

* **Canvas** – HTML елемент, що використовується для відтворення графіки засобами мов програмування, таких як JavaScript, використовується за необхідності завантаження на сторінки великого обсягу графічних об’єктів;
* **SVG** – масштабована векторна графіка, що базується на XML, на відміну від картинок формату PNG чи JPEGБ може бути масштабована без втрати якості зображення;
* **MathML** – мова розмітки математичних формул, викоритовується для відтворення рівнянь в мережі;

## **1.2. Мови Web-програмування, їх різновиди та функції**

**Веб-програмування** — це створення сайтів і програм, які працюють у мережі. Створюються спеціальні комп’ютерні програми — скрипти, які діляться на два типи: серверні і клієнтські. [5]

Сполучною ланкою між веб-розробником і сайтами є мова веб-програмування. Існує багато мов на яких пишуться і серверні, і клієнтські скрипти. Але деякі мови, навпаки, використовуються виключно для серверних чи клієнтських скриптів.

**Мова розмітки HTML** — основана на SGML текстова мова розмітки, призначена для маркування документів, що містять текст, зображення, гіперпосилання, тощо. HTML-документи лежать в основі Веб, і відображаються із допомогою веб-браузерів;[6]

**СSS** — це спеціальна мова (мова стилів), за допомогою якої описують вигляду документів (як і де відображати елементи веб-сторінки), написаних мовами розмітки даних. Найчастіше CSS використовується для документів, котрі розмічені мовою HTML, XHTML та XML;

Для програмування на стороні клієнта використовуються наступні мови і технології:

* **JavaScript** — об'єктно-орієнтована клієнтськамова програмування**,** що дозволяє реалізувати ряд складних рішень в web-документах. Вона допомагає зробити сторінки сайту більш інтерактивними, обробляє дії користувачів сайту. [7]
* **React** — це декларативна, ефективна і гнучка JavaScript-бібліотека, призначена для створення інтерфейсів користувача. [8]
* **VueJS** —це вдосконалений фреймворк JavaScript з відкритим вихідним кодом для розробки користувальницького інтерфейсу з використанням шаблону архітектури MVVM (Model-View-ViewModel). Це одна з найпопулярніших платформ для спрощення веб-розробки. [9]
* **AngularJS** — це MVW (Model-View-Whatever) JavaScript фреймворк від Google для створення складних SPA(Single Page) інтернет-програм, з використанням технології AJAX.[10]
* **jQuery** — фреймворк, розроблений на основі технології JavaScript, який допомагає спростити написання великих фрагментів коду. Бібліотека jQuery включає величезну кількість заготовлених функцій, які допомагають розробникам створювати інтерактивні елементи на сайті. Ця бібліотека працює з елементами DOM і дає можливість маніпулювати ними, а з іншого боку вона допомагає реалізувати API модулі для використання AJAX. **Основне призначення фреймворка jQuery** – програмування елементів веб-ресурсів та створення різних веб-додатків.[11]
* **AJAX** — підхід до побудови користувацьких інтерфейсів веб-застосунків, де веб-сторінка не перезавантажується, а у фоновому режимі відправляє запити на сервер і довантажує необхідні користувачу дані.[12]

Для програмування на стороні сервера використовуються наступні мови і технології:

* **Мова програмування PHP** — це сервероорієнтована скриптова мова, що використовується для розробки статичних веб-сайтів або динамічних веб-сайтів або веб-програм. [13]
* **Node.js** — платформа, побудована на середовищі JavaScript на Chrome для легкої побудови швидких та масштабованих мережевих програм. Node.js використовує керовану подіями модель неблокуючого вводу-виводу, що робить її легкою та ефективною, ідеально підходить для додатків у режимі реального часу, що працюють на розподілених пристроях.[14]
* **ASP.NET** — веб-платформа з відкритим кодом для створення веб-програм на платформі .NET (dotNET). Фреймворк побудований для роботи зі стандартним протоколом HTTP, який є стандартним протоколом, що використовується у всіх веб-програмах. Це розширення платформи .NET з додатковими інструментами та бібліотеками, спеціально створеними в Інтернеті, включаючи веб-програми та веб-сайти, що дозволяє розробникам створювати динамічні веб-програми, служби та сайти.[15]
* **Python** — це інтерпретована, об’єктно-орієнтована мова програмування високого рівня з динамічною семантикою. Високорівневі вбудовані структури даних у поєднанні з динамічним набором тексту та динамічним прив'язуванням роблять його дуже привабливим для швидкої розробки додатків, а також для використання в якості мови сценаріїв або склеювання для з'єднання існуючих компонентів разом. [16]

## **1.3. Різновиди ботів та їх застосування у Web.**

**Чат-бот** - це автоматизоване програмне забезпечення, створене для того, щоб допомогти користувачеві задовольнити його потреби при мінімальній взаємодії з системою та забезпечити найкращі результати із розумінням запитів користувача.[17]

Існує три види чат-ботів, що відрізняються між собою за функціональністю:

1. Прості чат-боти;
2. Розумні чат-боти;
3. Гібридні чат-боти.

Прості або керовані правилами чат-боти використовуються лише для вирішення простих питань, таких як: замовлення квитку у кінотеатр чи замовлення доставки на дім. Вони керуються деревом рішення, користувачеві надається набір наперед визначених опцій, що ведуть до бажаної відповіді. Вони прості у побудові та навчанні.[18]

В свою чергу, прості чат-боти поділяються на два види:

1. Чат-боти, що базуються на меню;
2. Чат-боти, що базуються на ключових словах.

Чат-боти на основі меню найчастіше зустрічаються сьогоднішній день. У більшості випадків ці чат-боти - це ієрархії дерев рішень, представлені користувачеві у вигляді кнопок. Ці чат-боти пропонують користувачеві вибрати кілька відповідей, щоб знайти точну відповідь. Хоча цих чат-ботів достатньо для відповіді на кілька основних питань, які складають 80% запитів підтримки; вони досить обмежені в більш просунутих сценаріях, в яких занадто багато змінних чи занадто багато знань, щоб передбачити, як користувач має отримати конкретну відповідь. [19]

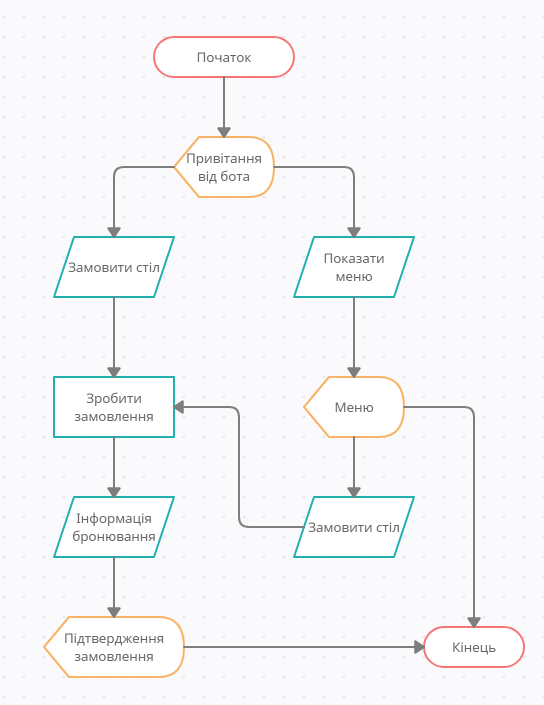


Рис. 1.1. Блок-схема роботи чат-бота, базованого на меню

На відміну від чат-ботів на основі меню, чат-боти на основі розпізнавання ключових слів розпізнають користувацький ввід і відповідають або намагаються відповідати належним чином. Ці чат-боти використовують настроювані ключові слова та ШІ щоб визначити, як надати належну відповідь користувачеві.

Наприклад, якщо користувач задає питання "Як налаштувати інтернет на пристрої TP-Link WR-841N?", бот, швидше за все, використовуватиме ключові слова "інтернет", "налаштувати" і " WR-841N ", щоб визначити найбільш підходящу відповідь.

Такі типи чат-ботів не підходять, коли їм доводиться відповідати на багато подібних питань. Чат-боти почнуть помилятись, коли між кількома схожими питаннями виникне надмірність ключових слів.[20]

Розумні чат-боти або чат-боти, керовані штучним інтелектом створені для симуляції подібної до людської взаємодії із користувачем. Завдяки штучному інтелекту вони здатні розуміти мову, наміри та почуття.

Чат-боти на базі штучного інтелекту є більш складними, досконалими, інтерактивними та персоналізованими, ніж попередньо розглянуті чат-боти.

Системи, засновані на машинному навчанні, можуть бути вражаючими, якщо розглянута проблема відповідає їх можливостям. За своєю природою така система навчається на зразках і попередньому досвіді. Але, для роботи навіть на найпростішому рівні, такі системи часто вимагають великих обсягів навчальних даних та висококваліфікованих спеціалістів. Крім того, чат-бот, що базується на ШІ, схожий на «чорний ящик». Якщо виникнуть проблеми із моделлю, буде важко визначити проблему, не кажучи вже про її оптимізацію та вдосконалення. Чат-боти на основі машинного навчання є непрактичним вибором для багатьох підприємств через кількість необхідних ресурсів в поєднанні з дуже вузьким діапазоном сценаріїв.[21]

Гібридні чат-боти можуть поєднувати у собі декілька різних видів чат-ботів, наприклад: чат-боти, що базуються на меню і чат-боти, що базуються на ключових словах, або чат-боти, керовані ШІ та один із видів простих чат-ботів.

Такі чат-боти ведуть розмову за визначеним сценарієм, як і в простих чат-ботах, але використовується ШІ для визначення користувацьких намірів, майнінгу цінних даних із користувацьких повідомлень.

Чат-боти можуть бути використані для:

* Користувацької підтримки 24/7: чат-боти – це програми, які можуть не зважати на час, і можуть проконсультувати в будь-який момент часу;
* Надання відповідей на часті питання: чат-боти здатні надавати відповіді в вигляді розмови, що більше подобається клієнтам;
* Пошук інформації у сервісах, наприклад: чат-бот Укрпошти або чат-бот Українського бюро кредитних історій;
* Сфера комунальних послуг: приом заявок на ремонт, надсилання інформації в випадку аварій і терміни їх вирішення, перевірка заборгованості;
* Сфера медичних послуг: отримати результати аналізів, записатися на прийом, викликати лікаря додому, пошук ліків та їх замовлення в аптеках. [22]

## **Висновки до Розділу 1**

**РОЗДІЛ 2.**

**ОРГАНІЗАЦІЯ РОЗРОБКИ WEB-ДОДАТКУ*.***

## **2.1. Огляд конструкторів чат-ботів**

**Конструктор чат-ботів** — це інструмент для створення віртуальних помічників, яка допомагає підприємствам автоматизувати комунікаційні процеси за заздалегідь визначеними сценаріями. Він дозволяє самостійно налаштовувати ботів без знання мови програмування, що робить його доступним кожній людині.

Він дозволяє надсилати новини, оновлення та нагадування; обробка замовлень; надання негайної підтримки клієнтам.

Конструктори чат-ботів підходять для будь-якого бізнесу: починаючи від невеликої сімейної компанії і закінчуючи великим підприємством. Вони також дешевші в порівнянні з чат-ботами, створеними розробниками з нуля.

Необхідно дослідити найбільш популярні конструктори чат-ботів, аби порівняти їх вартість та пропонований функціонал:

1. SendPulse. Ця багатоканальна маркетингова платформа дозволяє створити чат-бота для Facebook Messenger та Telegram. Безкоштовне надсилання до 10 000 повідомлень необмеженій кількості підписників. Щоб видалити логотип бренду з усіх віджетів і створити більше трьох чат-ботів, вам доведеться сплачувати 9,85 доларів на місяць. [23]
2. ManyChat. Платформа допомагає створювати цифрових асистентів для Telegram і Facebook Messenger. Про популярність цієї платформи свідчить офіційна статистика: завдяки ManyChat у світі створено 170 тисяч спамерських пошукових роботів. Створити чат-бота завдяки цій платформі можна за п’ять хвилин. Крім того, сервіс має базову безкоштовну версію. Такий бот буде займатися автоматичною розсилкою, у ньому можна створювати базу запитань і відповідей на них, текст, картинки, посилання.У ManyChat простий і інтуїтивно зрозумілий інтерфейс. Для роботи в ньому досить навичок звичайного користувача ПК. Вартість розширених пакетів стартує від $ 10.
3. FlowXO. Безкоштовна платформа для створення чат-ботів в Slack, Facebook і Telegram. Відкриває додаткові можливості після переходу на розширену версію. Бути програмістом зовсім не обов’язково. Відрізняється простотою у використанні, зрозумілим візуальним інтерфейсом. Базовий варіант дає можливість створити не більш ніж п’ять ботів на 500 взаємодій. Мінімальний тарифний план розширеної версії — $ 20.
4. Botsify. Платформа дає можливість самостійно сконструювати чат-бота з основним набором функцій і можливістю машинного навчання. Цифровий асистент зможе відповідати на питання, працювати з ключовими словами й підтримувати спілкування максимум із сотнею користувачів у місяць (у безкоштовній версії). Боти, створені завдяки Botsify, передають тексти, відео, фотографії й документи.
5. Chatfuel. Повністю безкоштовний сервіс для самостійного створення чат-бота в Facebook і Telegram. Цифрових асистентів, зроблених і налаштованих завдяки цій платформі, використовує канал Netflix, журнал Forbes і сервіс Uber. Користувачеві надається підбірка шаблонів, які легко налаштувати під себе без спеціальних знань. Цим сервісом користується 17 млн ​​осіб у світі. [24]

## **2.2. Особливості мови програмування Python**

Python - одна з найбільш динамічних та універсальних мов програмування, доступних сьогодні в галузі. З моменту свого заснування в 1990-х роках Python став надзвичайно популярним, і навіть сьогодні тисячі людей вивчають цю мову об’єктно-орієнтованого програмування. Якщо ви новачок у світі програмування, ви вже чули, який шум він створив останнім часом через особливості Python, і, мабуть, цікавитесь, чим ця мова програмування особлива.

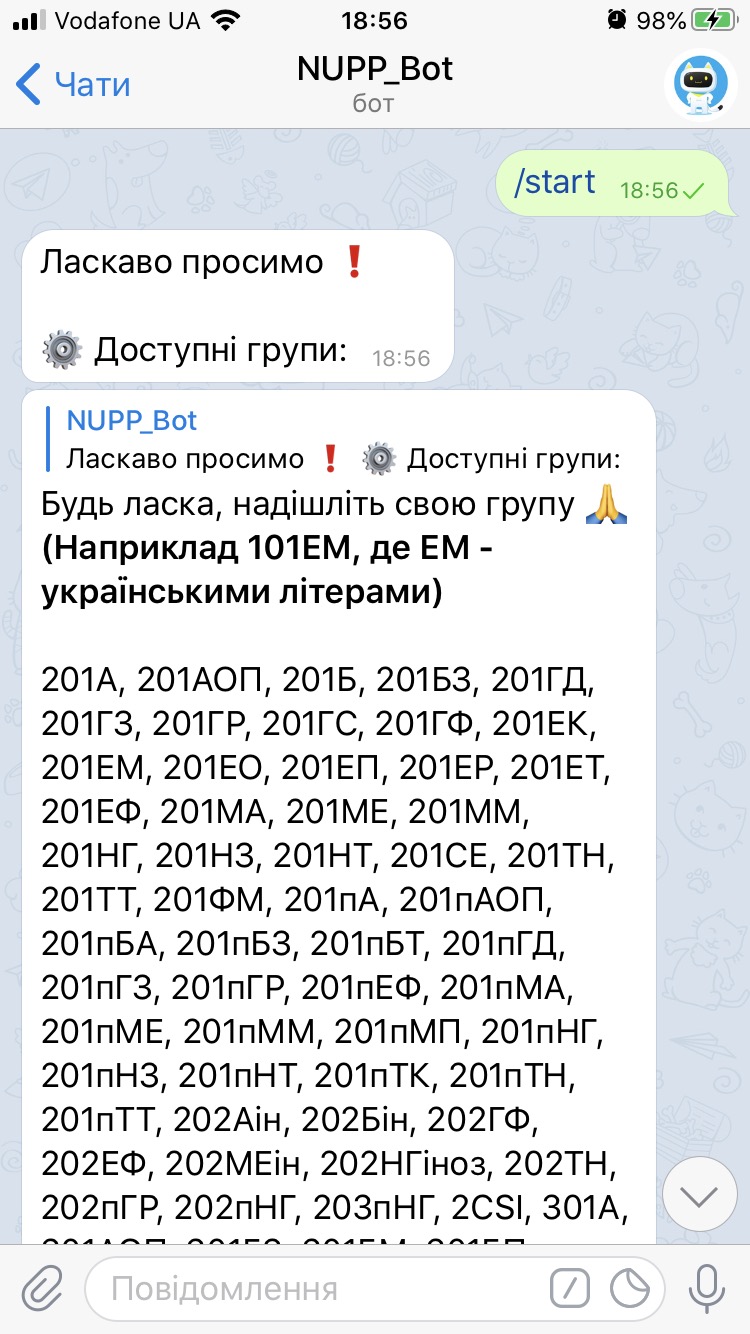
* 1. Легкий у вивченні та використанні: Python легко вивчити порівняно з іншими мовами програмування. Його синтаксис є простим, не використовується крапка з комою або фігурні дужки, відступ визначає блок коду. Це рекомендована мова програмування для початківців.
  2. Python може виконувати складні завдання, використовуючи кілька рядків коду. Простий приклад - програма “Hello world”, яку ви просто набираєте print ("Hello World"). Для виконання знадобиться лише один рядок, тоді як у мовах Java або C - кілька рядків.
  3. Python - інтерпретована мова; це означає, що програма Python виконується по одному рядку за раз. Перевага того, що мова інтерпретується, робить налагодження простим і портативним.
  4. Python є крос-платформенною мовою, а отже, однаково працює на різних платформах, таких як Windows, Linux, UNIX, Macintosh тощо. Це дозволяє розробникам створювати програмне забезпечення для декількох конкуруючих платформ, написавши програму лише один раз.
  5. Безкоштовний та відкритий код Python є у вільному доступі для всіх. Він знаходиться у вільному доступі на офіційному веб-сайті www.python.org. Він має велику спільноту по всьому світу, яка віддано працює над створенням нових модулів та функцій python. Будь-хто може зробити свій внесок у спільноту Python.
  6. Python підтримує об'єктно-орієнтовану мову, і існують концепції класів та об'єктів. Він підтримує успадкування, поліморфізм, інкапсуляцію тощо. Об'єктно-орієнтована процедура допомагає програмісту писати багаторазовий код та розробляти додатки з меншим кодом.
  7. Python є розширюваною мовою, що означає, що інші мови, такі як C / C ++, можуть бути використані для компіляції коду і використані в нашому коді в подальшому. Він перетворює програму в байт-код, і будь-яка платформа може використовувати цей байт-код.
  8. Python надає широкий спектр бібліотек для різних галузей розробки, таких як машинне навчання, веб-розробка, а також для скриптування. Існують різні бібліотеки машинного навчання, такі як Tensor flow, Pandas, Numpy, Keras, Pytorch та ін. Django, flask, pyramids - це популярний фреймворк для веб-розробки Python.
  9. Графічний користувальницький інтерфейс використовується для розробки робочого додатка. PyQT5, Tkinter, Kivy - це бібліотеки, які використовуються для розробки веб-додатків.
  10. Його можна легко інтегрувати з такими мовами, як C, C ++, JAVA тощо. Python запускає код рядком за рядком, як C, C ++ Java. Це полегшує налагодження коду.
  11. Код іншої мови програмування може використовуватись у вихідному коді Python. Ми можемо використовувати вихідний код Python і в іншій мові програмування. Він може вбудувати іншу мову в наш код.
  12. У Python нам не потрібно вказувати тип даних змінної. Коли ми присвоюємо змінному якесь значення, воно автоматично виділяє пам’ять для змінної під час виконання. Припустимо, нам присвоєно ціле значення 15 до x, тоді нам не потрібно писати int x = 15. Просто напишіть x = 15. [25]

## **2.3 Опис існуючих Telegram ботів.**

Необхідно провести аналіз існуючих чат-ботів з даної тематики та провести їх порівняльну характеристику. Даний процес дозволить виявити переваги та недоліки кожного аналогу для того, щоб виконати розробку власного чат-бота з урахуванням мінусів аналогів та вподобань замовника.

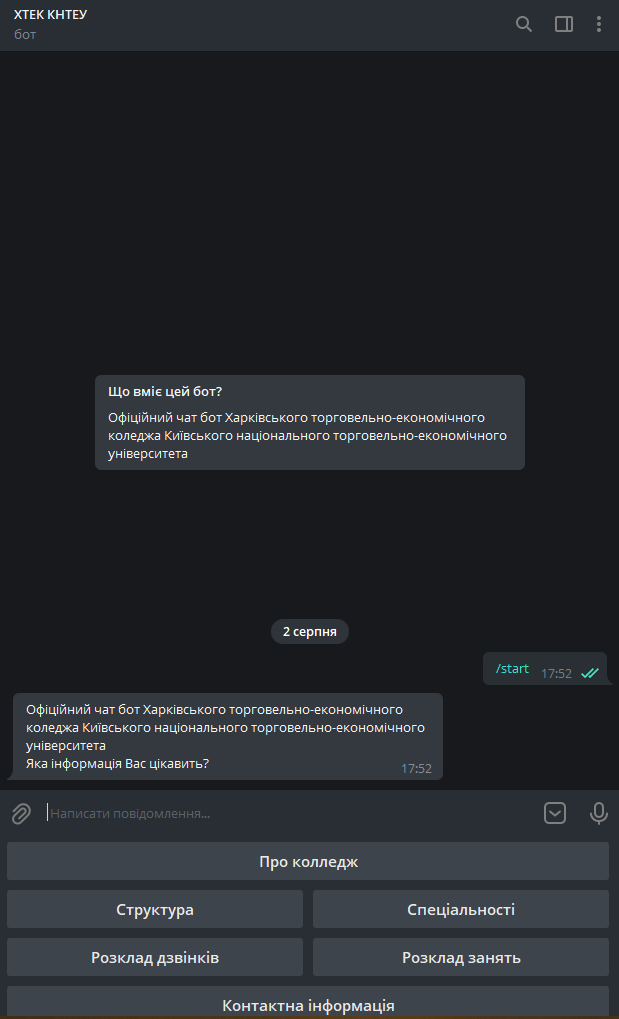
Першим знайденим аналогом є чат-бот Національного університету «Полтавська політехніка імені Юрія Кондратюка». Він призначений для пошуку розкладу занять на своєму телефоні і отримувати нагадування про пари. [26]

Даний чат-бот надає змогу отримувати розклад у трьох режимах: «сьогодні», «завтра», «поточний тиждень», Для цього необхідно ввести ботові назву своєї групи (наприклад, 101ЕМ, де ЕМ – українськими літерами), і одразу ж буде отримана відповідь.



Іншим знайденим чат-бот є чат-бот Харківського торговельно-економічного коледжу Київського національного торговельно-економічного університету.

Даний чат-бот розроблений на платформі Chatforma. На відміну від попереднього чат-боту, надає можливість ознайомитись не лише із розкладом занять, а й із розкладом дзвінків, контактною інформацією та іншими важливими даними.



Після того, як були знайдені існуючі чат-боти, необхідно провести порівняльну характеристику функціональних можливостей аналогів у порівнянні з власною розробкою.

|  |  |  |  |
| --- | --- | --- | --- |
| Назва критерію | Назва чат-боту | | |
| NUPP\_bot | ХТЕК КНТЕУ | Власна розробка |
| * + 1. Наявність переліку команд керування | + | + | + |
| * + 1. Відображення розкладу занять у зручному вигляді | + | - | + |
| * + 1. Додаткові функції налаштування розкладу | + | - | + |
| * + 1. Додаткові пункти контекстного меню | - | + | + |

Таблиця 1.1 – Порівняльна характеристика існуючих аналогів з власною розробкою

Після проведення порівняльної характеристики існуючих чат-ботів інформаційної підримки освітнього процесу та власної розробки «EatArt» було прийнято рішення про реалізацію чат-бота «EatArt», оскільки розглянуті аналоги не мають у своєму складі цілісної картини функціоналу, який би задовольняв потреби замовника.

У аналогу від ХТЕК КНТЕУ відсутня функція зручного відображення розкладу занять, оскільки розклад відображається у вигляді фотографії . До того ж, аналоги NUPP\_bot та ХТЕК КНТЕУ не надають користувачу додаткових пунктів вибору у контекстному меню, корисних у процесі навчання. Тому було прийнято рішення про власну розробку чат-бота «EatArt», який би містив у своєму складі переваги розглянутих аналогів та усунуті їх недоліки.

## **Висновки до Розділу 2**

# **РОЗДІЛ 3.**

# **ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОГО ЧАТ-БОТА ДЛЯ ПІДТРИМКИ ОСВІТНЬОГО ПРОЦЕСУ В СЕРЕДОВИЩІ TELEGRAM**

## **3.1. Розробка структури інформаційного чат-бота для підтримки освітнього процесу в середовищі Telegram**

Telegram боти дозволяють зняти з себе багато рутинних справ, які користувачеві доводилось колись здійснювати вручну. Вони можуть виконувати безліч задач: автоматизувати прийом заявок, видача інформації користувачеві за запитом, повідомлення користувача про якісь події, наприклад про погодні умови. Саме тому Telegram бот є корисним помічником для будь-якого студента університету.

Структура чат-бота @ «вставити назву» виглядає наступним чином:

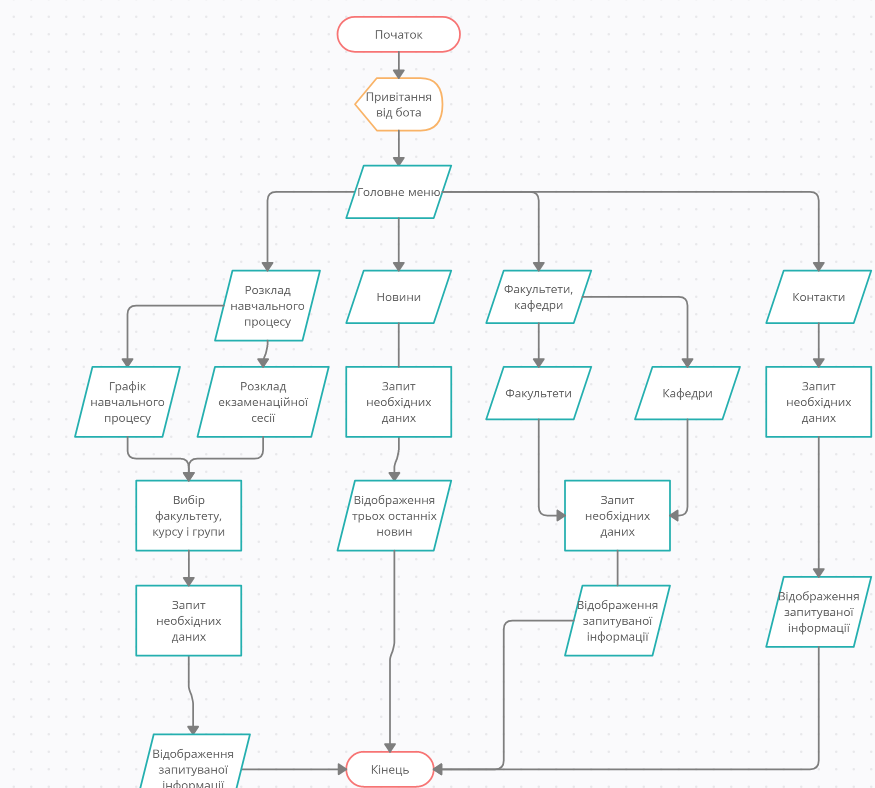


Рис. 3.1 Структура Telegram бота КНТЕУ [авторська розробка]

Після того, як користувач натисне команду «/start», чат-бот розпочне свою роботу. Складовими елементами є:

* Розклад навчального процесу;
* Новини;
* Факультети, кафедри;
* Контакти.

Користувач чат-боту, натискаючи на певну кнопку меню, виконує запит на сервер, де знаходиться бот, обробивши запит, бот відправляє відповідь через сервери Telegram користувачу.

Отже, затвердження структури є необхідною складовою для розробки Telegram боту. Саме нею розробники керуються на всіх етапах створення боту.

## **3.2. Розробка моделі функціонування інформаційного чат-боту для підтримки освітнього процесу в середовищі Telegram**

**Модель** – деякий матеріал або подумки представлений об’єкт або явище, що заміщає оригінальний об’єкт або явище, зберігаючи тільки деякі важливі його властивості, наприклад, в процесі пізнання (споглядання, аналізу та синтезу) або конструювання. [27]

**Функціональна модель** дозволяє виконати детальний аналіз роботи системи, розглянути і проаналізувати напрямки удосконалення роботи. У технічних системах дана модель дозволяє вирішити проблеми раціонального проектування складних технічних систем, зробити їх більш дешевими, простими та функціонально спрямованими.[28]

Для створення моделі функціонування чат-боту було використано **AllFusion Process Modeler r7**, де для моделювання чат-боту було використано діаграми IDEF0.

**Основа методології IDEF0** – графічна мова опису системи. Спочатку проводиться опис системи в цілому і її взаємодії з навколишнім світом (контекстна діаграма), після чого проводиться функціональна декомпозиція - система розбивається на підсистеми і кожна підсистема описується окремо (діаграми декомпозиції). За допомогою блоків на даній діаграмі зображуються процеси, які з’єднуються стрілками – зв’язки між процесами. На графіку IDEF0 може бути 5 типів стрілок:

* 1. Вхід - об'єкти, що використовуються як вхідні дані для процесу. Допускається, що робота може не мати жодної стрілки входу. Стрілка входу направлена в лівий бік процесу.
  2. Управління - інформація, керуюча процесами роботи. Керуючі стрілки несуть інформацію, яка вказує, що повинен виконувати процес і яким чином. Кожен процес повинен мати хоча б одну стрілку управління, яка направлена у верхню грань.
  3. Вихід - об'єкти, в які перетворюються входи. Кожен процес повинен мати хоча б одну стрілку виходу, яка виходить з правої межі процесу.
  4. Механізм - ресурси, які виконують роботу. Стрілка механізму малюється направленою в нижню межу процесу.
  5. Виклик - спеціальна стрілка, що вказує на іншу модель роботи. Стрілка виклику направлена з нижньої частини процесу і використовується для вказівки того, що деяка робота виконується за межами модельованої системи. Не є обов’язковою, використовується не часто.

На контекстній діаграмі (рис.3.2) вхідними даними є запит користувача до чат-боту, база розкладів навчального процесу і дані про групу користувача.

Стрілки управління відображають документацію, необхідну для виконання процесу, в даному випадку це документація Python і документація TelegramBotAPI[29].

Стрілки механізмів представляють собою виконавців, на діаграмі зображені як мова програмування Python, месенджер Telegram та користувача сервісу.

Виходом є результат виконання процесу, а саме: виконання основної функції даного чат-боту: надання правильно підібраного розкладу.

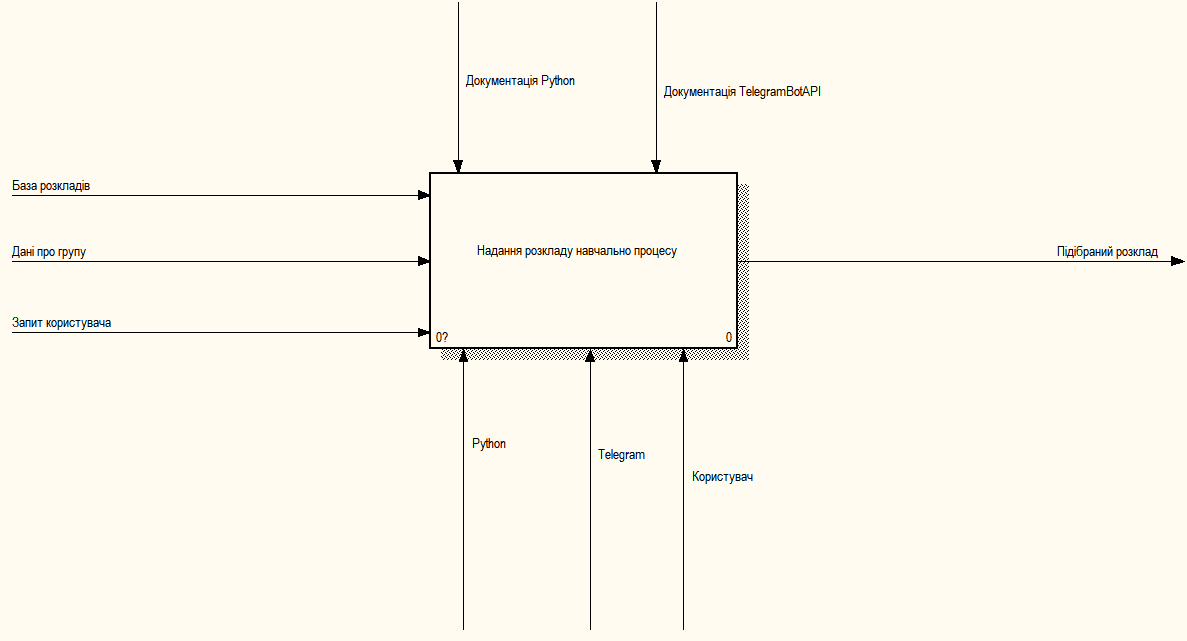


Рис. 3.2. Контекстна діаграма процесу надання розкладу(авторська розробка)

Вхідними даними є запит користувача на отримання розкладу, дані про його групу, база розкладів університету. Після отримання запиту користувач перенаправляється до пункту введення необхідних даних. Після отримання необхідних даних виконується пошук необхідного розкладу в базі розкладів університету. Якщо необхідних даних немає, з’являється сповіщення про помилку. Якщо ж необхідні дані було знайдено, виконується їх обробка для відображення у найбільш зручному для сприйняття користувачем вигляді.

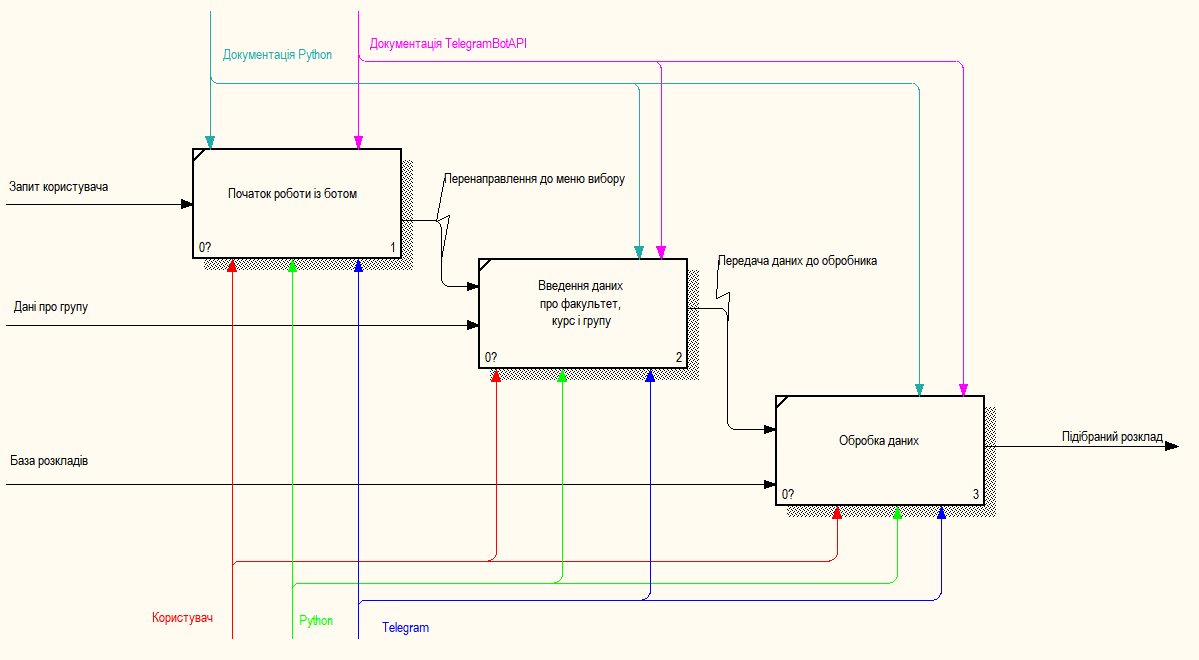


Рис. 3.3. Діаграма декомпозиції першого рівня(авторська розробка)

## **3.3. Пpoгpaмнa peaлiзaцiя проекту**

Для того щоб створити «КНТЕУbot» знайдемо BotFather у пошуку Telegram (рис. 3.4).

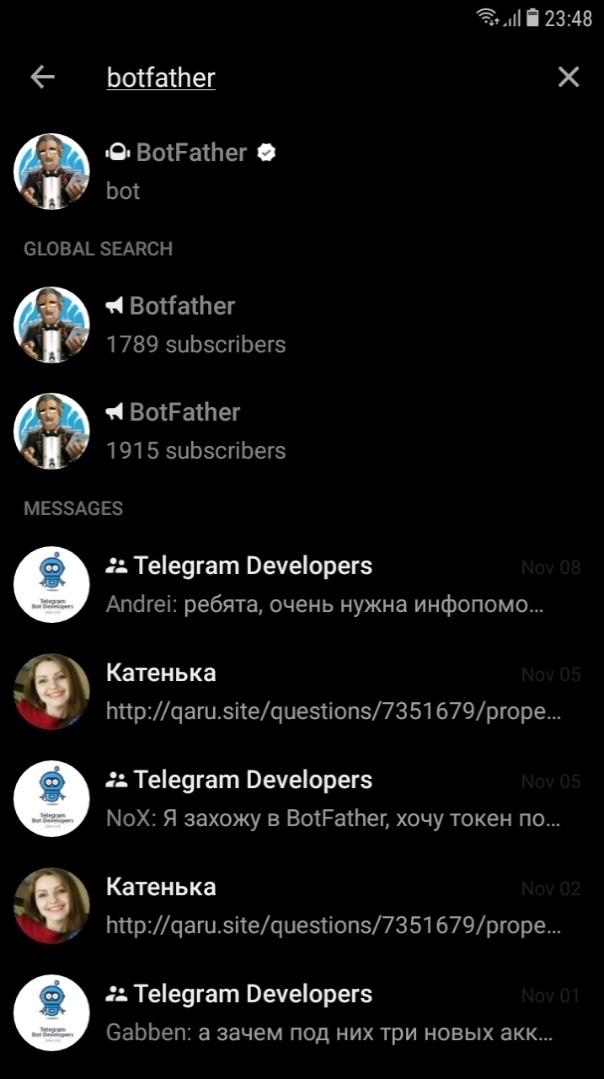


Рис. 3.4 BotFather

Напишемо /newbot, після вкажемо ім’я «назва», наступний крок вказати посилання. Якщо все зроблено вірно, BotFather відповість вам наступним чином (рис. 3.5)., а саме: згенерує посилання на чат-бота та надасть токен(ключ), який необхідно вказати у програмному коді, аби була можливість під’єднати код до чат-бота.

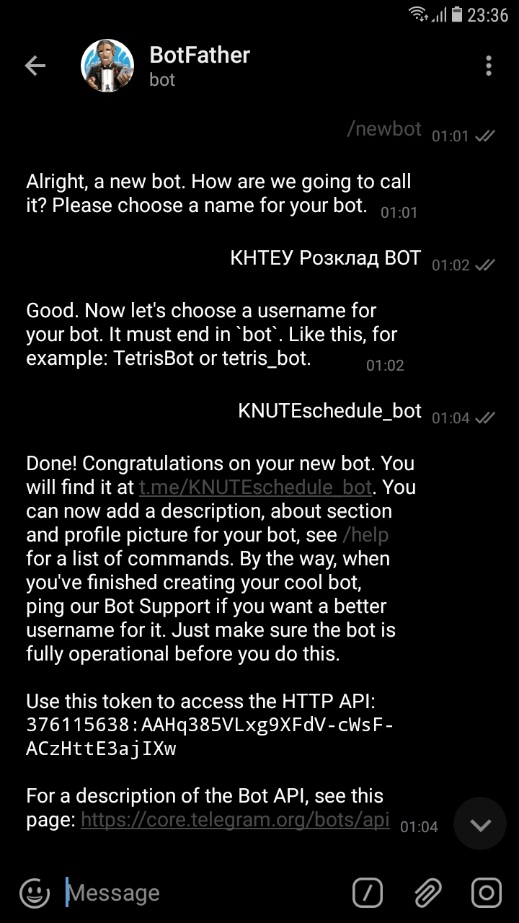


Рис. 3.5 Повідомлення від BotFather

Щоб почати користуватися нашим ботом його потрібно знайти в пошуку Telegram (рис.3.6).

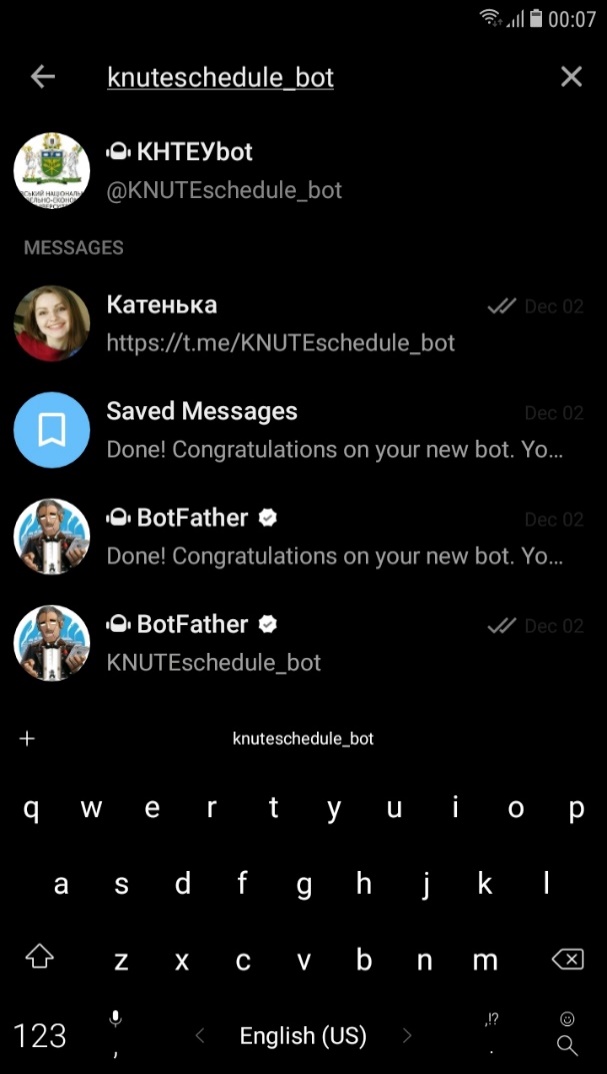


Рис. 3.6 Пошук «КНТЕУ Розклад bot»

Створимо новий проект в редакторі коду Visual Studio Code[30]. Для роботи бота використовується декілька файлів, які необхідні для його коректної роботи, а саме:

* main.py, що містить в собі основний код бота;
* get\_schedule.py, що містить програмний код, необхідний для отримання посилань на розклади груп;
* get\_news.py – включає код, необхідний для запиту новин із сайту університету;
* get\_kafedra.py – містить код, необхідний для пошуку посилань на факультети і кафедри університету.

Крім того, для роботи чат-боту необхідно імпортувати декілька бібліотек:

* pyTelegramBotAPI;
* pandas [31];
* NumPy [32];
* dataframe\_image [33];
* requests [34];
* BeautifulSoup[35];

Для того, щоб під’єднати програмний код для чат-бота, необхідно задати у коді програми токен, який був виданий BotFather. За дану функцію відповідає наступна команда:

bot = telebot.TeleBot('1796930178:AAFZUChk3App456JxIjSQkvKOzrVjAIApi0')

На рис 3.7 зображено стартовий екран бота на якому описано його можливості. Почати роботу можливо після натискання на кнопку «Start». За обробку цієї команди відповідає декоратор @bot.message\_handler(commands=['start']).

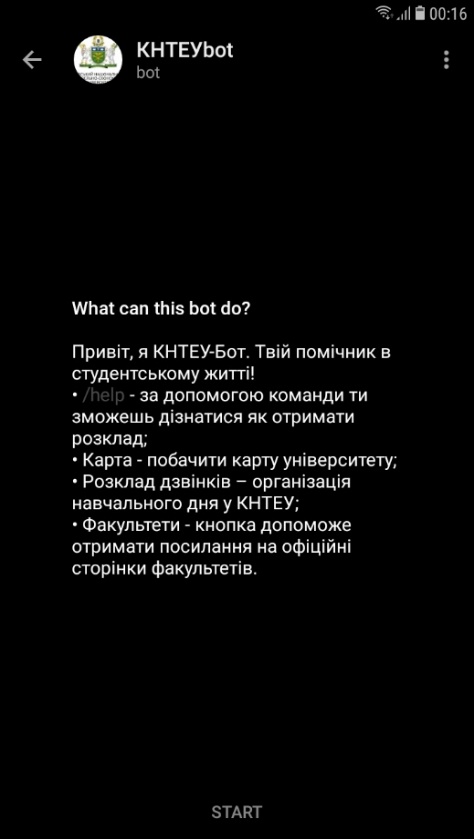


Рис. 3.7 Пошук «КНТЕУ Розклад bot»

Для того, щоб наш бот не зупинявся та постійно обробляв запити використовуємо функцію bot.polling(none\_stop=True, interval=0).

На рис. 3.8 зображено стартове меню бота з основними функціональними кнопками. За відображення меню відповідає об’єкт keyboardmain = types.InlineKeyboardMarkup(row\_width=2), що включає в себе об’єкти InlineKeyboardButton. Кожній із кнопок меню відповідає параметр text і callback\_data.

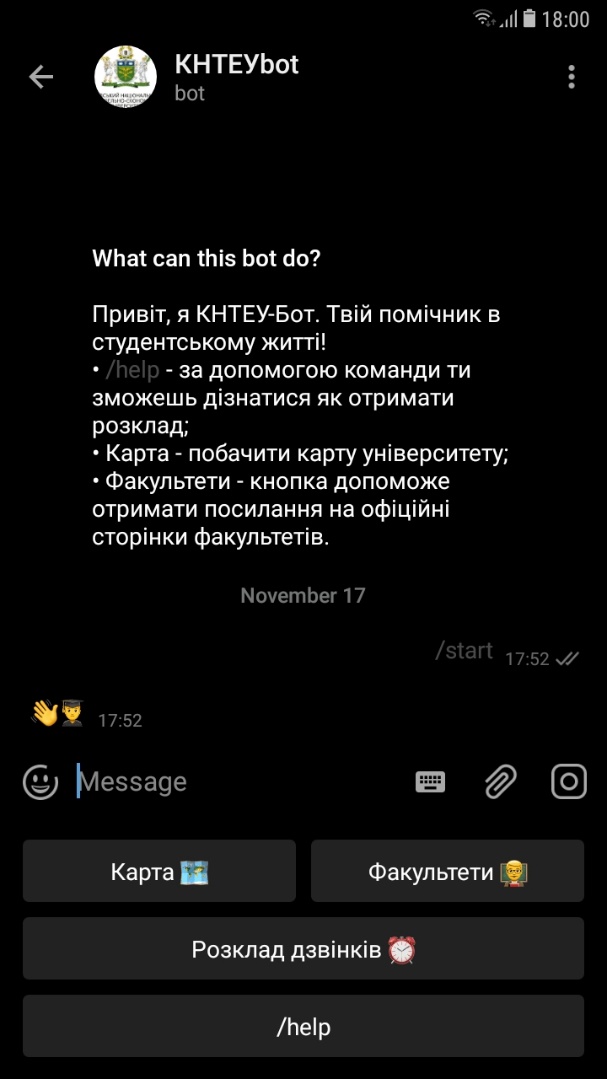


Рис. 3.8. Стартове меню «КНТЕУ Розклад bot»

Параметр callback\_data включає в себе спеціальне значення (data), за яким ваш додаток розпізнає, яка кнопка натиснута і що треба зробити. Варто також відзначити, що, на відміну від звичайних кнопок, натискання на колбек-кнопку дозволяє зробити практично що завгодно, від замовлення піци до перезавантаження сервера.

## **Висновки до Розділу 3**

# **ВИСНОВОК**

# **СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. [Web Technologies: A Journey From HTML to Web 3.0 (goodcore.co.uk)](https://www.goodcore.co.uk/blog/web-technologies/)
2. All About Web Technologies [Електронний ресурс]. – Режим доступу: <https://www.lsnsoft.com/all-about-web-technologies/>
3. **Робота із браузером Іnternet Explorer (Mozilla Firefox).** [Електронний ресурс]. – Режим доступу: <http://lib.mdpu.org.ua/e-book/oi/pract1.htm>
4. [Що таке база даних? | Кафедра АПЕПС ТЕФ КПІ — програмна інженерія (kpi.ua)](http://apeps.kpi.ua/shco-take-basa-danykh)
5. [Веб-програмування | Кафедра АПЕПС ТЕФ КПІ — програмна інженерія (kpi.ua)](http://apeps.kpi.ua/web-programuvania)
6. [HTML — мова розмітки гіпертексту | Портал знань, портал знаний, дистанційне навчання (znannya.org)](http://www.znannya.org/?view=html)
7. [JavaScript - Клієнтською мова програмування, що робить сторінки сайту інтерактивними (astwellsoft.com)](https://astwellsoft.com/uk/blog/tehnology/javascript.html)
8. [Посібник: знайомство з React – React (reactjs.org)](https://uk.reactjs.org/tutorial/tutorial.html#what-is-react)
9. [What is VueJS (newline.tech)](https://newline.tech/what-is-vuejs/)
10. [Особливості фреймворку AngularJS (dzudzylo.com)](https://dzudzylo.com/javascript/osoblyvosti-frejmvorku-angularjs.html)
11. [jQuery - Фреймворк дозволить швидко та якісно створювати інтерактивні сторінки сайту (astwellsoft.com)](https://astwellsoft.com/uk/blog/tehnology/jquery.html)
12. [Технологія Ajax і сайти в концепції Web 2.0 | Портал знань, портал знаний, дистанційне навчання (znannya.org)](http://www.znannya.org/?view=AJAX-technology)
13. [What is PHP? Write your first PHP Program (guru99.com)](https://www.guru99.com/what-is-php-first-php-program.html)
14. [Node.js - Introduction - Tutorialspoint](https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm)
15. [What is ASP.NET? | The Open Source Web Framework (umbraco.com)](https://umbraco.com/knowledge-base/aspnet/)
16. [What is Python? Executive Summary | Python.org](https://www.python.org/doc/essays/blurb/)
17. Ali Abdel Aal. Building Your First Telegram Bot: A Step by Step Guide [Електронний ресурс]. – Режим доступу: <https://www.toptal.com/python/telegram-bot-tutorial-python>
18. Konstantin Sadekov. Types of Chatbots. Rule-Based Chatbots vs AI Chatbots. [Електронний ресурс]. – Режим доступу: <https://mindtitan.com/resources/guides/chatbot/types-of-chatbots/>
19. The 3 Kinds of Chatbots You'll Meet. [Електронний ресурс]. – Режим доступу: <https://www.goboomtown.com/blog/3-kinds-chatbots-youll-meet>
20. Casey Phillips. The 3 Types of Chatbots & How to Determine the Right One for Your Needs. [Електронний ресурс]. – Режим доступу: <https://chatbotsmagazine.com/the-3-types-of-chatbots-how-to-determine-the-right-one-for-your-needs-a4df8c69ec4c>
21. Chatbots: The Definitive Guide (2020) – Artificial Solution. [Електронний ресурс]. – Режим доступу: <www.artificial-solutions.com>
22. [What are chatbots used for? 7 examples in everyday life (unblu.com)](https://blog.unblu.com/en/what-are-chatbots-used-for-7-examples-in-everyday-life)
23. [What is Chatbot Builder: Definition, Video, Tips - Definition | SendPulse](https://sendpulse.com/support/glossary/chatbot-builder)
24. [Безкоштовні сервіси для швидкого створення чат-бота | AI CONFERENCE KYIV 2020](https://aiconference.com.ua/uk/news/besplatnie-servisi-dlya-bistrogo-sozdaniya-chat-bota-94385)
25. [Features of Python | Top 10 Features of Python programming | Edureka](https://www.edureka.co/blog/python-features/) [Python Features - javatpoint](https://www.javatpoint.com/python-features)
26. [Розклад занять завжди в телефоні: університет запустив на Telegram чат-бот для студентів (nupp.edu.ua)](https://nupp.edu.ua/news/rozklad-zanyat-zavzhdi-v-telefoni-universitet-zapustiv-na-telegram-chat-bot-dlya-studentiv.html).
27. [Що таке модель? Учасники процесу моделювання. Види моделей та вимоги до них | Kursak.com - здаємо разом!](https://www.kursak.com/shcho-take-model-uchasnyky-protsesu-modeliuvannia/)
28. [СисАнализ9-16н (core.ac.uk)](https://core.ac.uk/download/pdf/11320265.pdf)
29. <https://github.com/python-telegram-bot/python-telegram-bot>
30. [Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/)
31. [pandas - Python Data Analysis Library (pydata.org)](https://pandas.pydata.org/)
32. [NumPy](https://numpy.org/)
33. [dexplo/dataframe\_image: A python package for embedding pandas DataFrames as images into pdf and markdown documents (github.com)](https://github.com/dexplo/dataframe_image)
34. [requests · PyPI](https://pypi.org/project/requests/)
35. [Beautiful Soup Documentation — Beautiful Soup 4.9.0 documentation (crummy.com)](https://www.crummy.com/software/BeautifulSoup/bs4/doc/)

[Shcherbak\_bac\_rob.pdf;jsessionid=3CBE5A57C8F0DA976A14ACDC0EC37DEF (sumdu.edu.ua)](https://essuir.sumdu.edu.ua/bitstream-download/123456789/79600/1/Shcherbak_bac_rob.pdf;jsessionid=3CBE5A57C8F0DA976A14ACDC0EC37DEF)

https://habr.com/ru/post/346606/?\_ga=2.120166474.311363403.1616608844-11755294.1590517715