МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

СТЕЦЮК СЕРГІЙ ОЛЕГОВИЧ

Допускається до захисту:
Завідувач кафедри
інформаційних технологій,
доктор технічних наук, доцент
_____Т. В. Нєскородєва
«___»_____20__р.

РОЗРОБКА ТЕЛЕГРАМ-БОТУ ДЛЯ ПОШУКУ МІЖМІСЬКИХ АВТОБУСНИХ РЕЙСІВ ВІННИЦЬКОЇ ОБЛАСТІ

Спеціальність 122 Комп'ютерні науки

Кваліфікаційна (бакалаврська) робота

Керівник:
Бабаков Р.М., доцент кафедри
інформаційних технологій,
д.т.н, доцент

Оцінка:____/____/_____
(бали за шкалою СКТС/за національною шкалою)

Голова ЕК:_____
(підпис)

Вінниця – 2022

АНОТАЦІЯ

Стецюк С. О. Розробка телеграм-боту для пошуку міжміських автобусних рейсів . Спеціальність 122 «Комп'ютерні науки», спеціалізація «Інформаційні технології»

Донецький національний університет імені Василя Стуса, Вінниця, 2022.

У кваліфікаційній (бакалаврській) роботі досліджені сучасні методи розробки телеграм-ботів. Розгляд аналогів, пошук та вивчення представлених телеграм-ботів, які мають функції пошуку. Розробка ботів, котрі виконують команди. Розробка телеграм-боту з ціллю надання зручного та швидкого пошуку потрібного транспорту за маршрутом.

Ключові слова: телеграм, бот, аналіз, підбір, javascript, сучасні технології, розробка.

51 с., 29 рис., 29 джерел.

ABSTRACT

Stetsiuk SO Development of telegram-bot for search of intercity bus routes. Specialty 122 "Computer Science", specialization "Information Technology". Vasyl Stus Donetsk National University, Vinnytsia, 2022.

In the qualification (bachelor's) work the modern methods of telegram-bots development are investigated. Consideration of analogues, search and study of the presented telegram bots, which have search functions. Development of bots that execute commands. Development of a telegram-bot in order to provide a convenient and fast search for the desired transport along the route.

Key words: telegram, bot, analysis, selection, modern technologies, development.

51 p., 29 draw., 29 source.

ЗМІСТ

ВСТУП……...5

Розділ 1. …… 6

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ, ПОСТАНОВКА ЗАДАЧІ ТА ОГЛЯД ІСНУЮЧИХ АНАЛОГІВ7

1.1 Актуальність теми……...6

1.2 Розгляд аналогів……..7

1.3 Постановка задачі……...11

Розділ 2. …… 12

АНАЛІЗ ТА ВИБІР АКТУАЛЬНИХ ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ……12

2.1 Вибір програмних засобів розробки……12

Розділ 3...... 27

РЕАЛІЗАЦІЯ ТА СТВОРЕННЯ ТЕЛЕГРАМ-БОТА ……27

3.1 Створення телеграм-бота……...27

3.2 Розробка телеграм-бота……..33

3.3 Логіка роботи телеграм-боту……40

ВИСНОВОК**…**……45

СПИСОК ВИКОРИСТАНИХ ПОСИЛАНЬ**…**……...46

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ,СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

JS – JavaScript (Мова програмування JavaScript);

DOM – Document Object Model (Об’єкт модель документа)

JSON – JavaScript Object Notation (Запис об’єктів JavaScript)

WS – WebStorm (Інтегроване середовище розробки)

API – Application programming interface (Програмний інтерфейс)

NPM – Node Packet Manager (Менеджер пакетів Node.js)

JSON – JavaScript Object Notation (Запис об’єктів JavaScript)

ВСТУП

Сучасний світ не стоїть на місці, технології досить швидко розвиваються майже кожного дня реалізуються нові ідеї, більшість з них досить сильно впливають на людство, таким чином це впливає на ІТ галузі.

Не так давно ще були популярні додатки або програми на комп'ютері, завдяки яким можна було спілкуватись з рідними та друзями, або ж для пошуку нових знайомств. На даний момент лідерами стали додатки з чат-ботами, які досі не вичерпали свій потенціал та мають досить великі перспективи для розвитку в майбутньому.

Перш за все необхідно зрозуміти, що собою представляє чат-бот. Чат-бот- це додаток, який дозволяє користувачам взаємодіяти з різноманітними сервісами, при цьому витративши на це мінімальну кількість часу. Чат-бот виступає в ролі помічника, який спілкується з користувачами через повідомлення і має достатньо багато функцій. Тобто можна отримати інформацію, написавши чат-боту спеціальну команду, яку можна вибрати з переліку.

Тема бакалаврської роботи – «Розробка телеграм-боту для пошуку міжміських автобусних рейсів». Вказана тема є досить актуальною, адже технології не стоять на місці та постійно розвиваються.

Мета бакалаврської роботи – створення телеграм-боту для пошуку міжміських автобусних рейсі . За допомогою якого, будь-який користувач зможе знайти потрібний йому рейс з одного міста в інше.

Розділ 1.

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ, ПОСТАНОВКА ЗАДАЧІ ТА ОГЛЯД ІСНУЮЧИХ АНАЛОГІВ

1.1. Актуальність теми

У сучасному світі мобільні пристрої досить сильно розвинені, тому є великий попит на додатки для таких девайсів. Досить велику популярність отримали мессенджери та соціальні мережі, якими в наш час користується майже кожен. Одним із найпопулярніших мессенджерів є Telegram, який простий, швидкий та безпечний додаток для обміну повідомленнями. Telegram входить в десятку найбільш завантажуваних програм у світі, ним користується понад 500 мільйонів людей.

В цій соціальній мережі одним із головних плюсів для бізнесу та організаційної роботи є так звані боти. Майже всі сучасні компанії та онлайн-магазини використовують саме ботів для вирішення таких задач як: дізнатись певну інформацію або зробити замовлення.

Боти створені задля спрощення деяких поставлених задач не тільки для крупних компаній а й для малого бізнесу. За рахунок ботів можна скоротити кількість персоналу та зменшити ймовірність непорозумінь, адже в любий момент часу керівник зможе перевірити потрібну йому інформацію. Таким чином можна зменшити витрати на зарплатню, оренду офісних місць для працівників.

Telegram доступний на всіх мобільних платформах.

1.2 Розгляд аналогів

1.2.1 Airtrack_bot

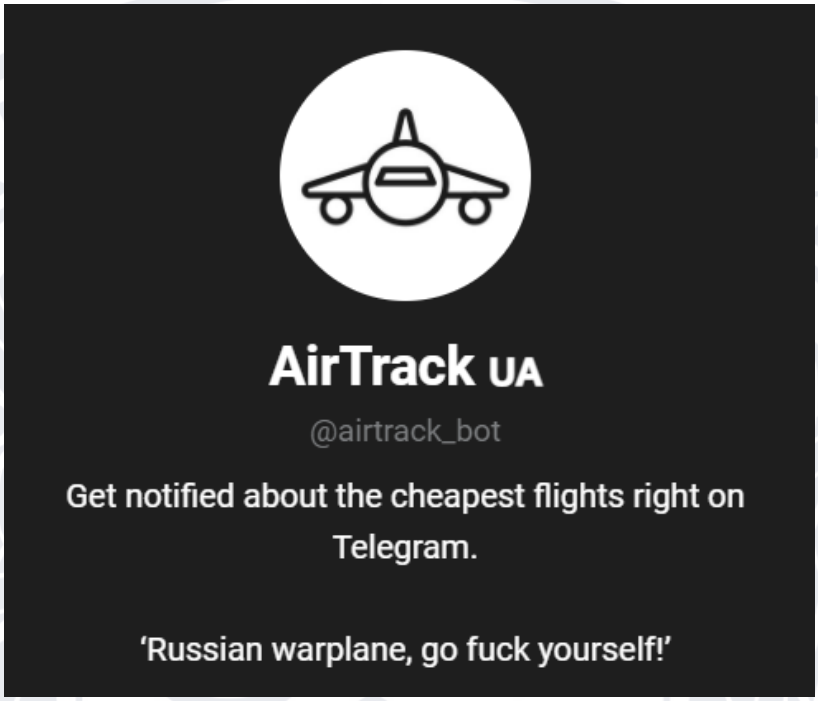


Рисунок. 1.1 – телеграм-бот AirTrack

Помічник у пошуку авіаквитків з будь-якого міста світу до іншого. Як і в будь-якому боті для початку потрібно натиснути кнопку «Старт»

Бот пропонує вибрати мову, на якій користувачу буде зручно отримувати інформацію. Після чого потрібно вибрати мову та валюту. Далі потрібно вказати пункт відправлення ,місто прибуття та дату прибуття. Згодом після обробки даних користувач отримує інформацію щодо вибраного маршруту, якщо ж такого не знайшлося то буде відправлене повідомлення про відсутність авіарейсу.

Серед AirTrack переваг слід виділити:

1. Можливість вибрати мову, валюту;
2. Зручний та зрозумілий функціонал
3. Швидка робота бота

Із недоліків в AirTrack можна відмітити те, що безкоштовна версія має урізаний функціонал.

1.2.2 RailwayBot



Рисунок. 1.2 – телеграм-бот RailwayBot

Даний телеграм-бот виконує пошук залізничних квитків.

Для того, щоб почати користуватись ботом, потрібно ввести свій номер телефону, після чого він активується та далі виконує команди користувача.

Слід відмітити, що бот також може повернути кошти за квиток та сповістити вас якщо по вибраному вами маршруту з’явиться квиток на потяг.

Переваги **RailwayBot:**

1. Зручне меню
2. Можливість повернути квиток
3. Функція моніторингу
4. Безкоштовне користування

Серед недоліків **RailwayBot** лише те, що обробка інформації займає близько 1-2 хвилин, через великий діапазон вибору квитків.

1.2.3 Depovnua_bot

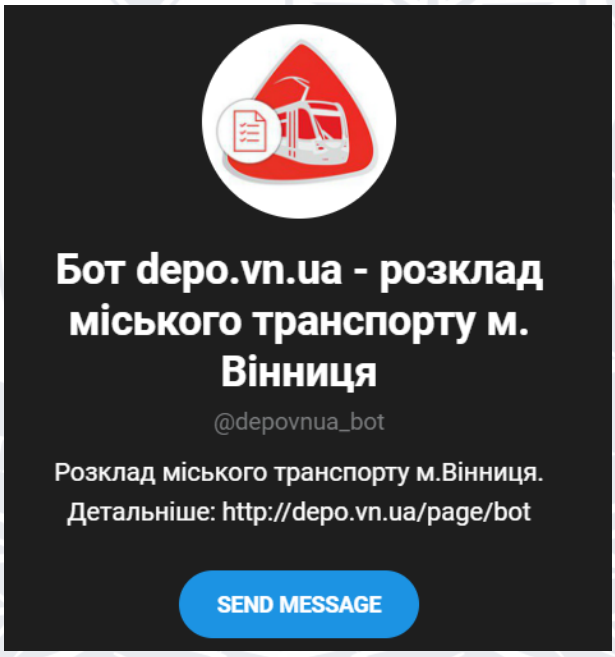


Рисунок. 1.3 – телеграм-бот depo.vn.ua

Бот, який показує розклад руху міського транспорту, досить зручно, зважаючи на те, що деколи на зупинках час прибуття транспорту не вірний через не актуальність інформації.

Через конфіденційні правила в Telegram в ботах ділитись геолокацією не

скрізь можна, тому лише вказавши вулицю, по якій вам потрібен розклад руху, зможете дістатись до потрібної вам локації. Також слід відмітити, що є можливість побачити рух транспорту під час святкових днів.

Інформація щодо рейсів транспорту подана досить містким повідомленням, тому юзерам які не змогли її розтлумачити , є меню «Допомога» , де є можливість розібратись з потрібним їм транспортом.

1.3 Постановка задачі

Мета бакалаврської роботи полягає в аналізі подібних існуючих аналогів телеграм-ботів, вибір актуальних технологій для розробки власного телеграм чат-боту.

Для досягнення визначеної мети необхідно:

- пошук аналогів
- визначення їх переваг та недоліків
- обрати хостинг сервер
- розробити базу даних
- вибір середовища розробки
- реалізація функції проїзду з пересадками
- додати інформацію про рейси
- створити власний повноцінний телеграм-бот

Розділ 2.

АНАЛІЗ ТА ВИБІР АКТУАЛЬНИХ ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Вибір програмних засобів розробки

В даному розділі розглянуті програмні засоби та технології, які були використані для розробки боту, вибрано середовище розробки, бібліотеки API Telegram та шляхи реалізації заданих функцій.

2.1.1 JavaScript

JavaScript або JS — це мова програмування, яка працює у браузерах. Він забезпечує інтерактивність веб-сайтів, оновлення вмісту в реальному часі та параметри анімації. Спочатку він був створений для того, щоб веб-сторінки оживали.

JavaScript — це код, який можна написати в такому простому, як текстовий редактор. Це одна з найпростіших мов програмування для вивчення завдяки своєму простому та зрозумілому коду. Це також вважається мовою Інтернету. Його можна виконувати як на стороні клієнта, так і на стороні сервера, або на будь-якому пристрої, який має механізм JavaScript.

Ці механізми JavaScript мають різні назви на різних платформах. Chrome і Opera використовують V8, Firefox використовує SpiderMonkey, а Safari використовує SquirrelFish. Незалежно від того, як називається двигун, двигун JavaScript завжди читатиме мову JavaScript.

Однак JavaScript не слід плутати з Java, яка є іншим типом мови програмування. Спочатку JavaScript називався LiveScript. Але через тодішню славу Java її перейменували в JavaScript, позиціонуючи її як молодшого брата Java.

Кожен, хто коли-небудь користувався комп'ютером або смартфоном, стикався з JavaScript. Це те, що робить веб-сторінки більш інтерактивними та сприяє більш плавному взаємодії з користувачем, що може принести користь бренду компанії. Крім того, майже всі смартфони використовують програми, створені за допомогою цієї мови програмування.

JavaScript підтримується всіма основними браузерами і має повну інтеграцію з HTML і CSS. Без цього веб-сайти були б просто текстом і зображеннями, Google Maps — просто базовою картою, а Facebook виглядав би як газета. JavaScript оживає сторінки, і це те, що робить Інтернет таким, яким він є сьогодні.

Спочатку створений для використання у браузерах, зараз JavaScript використовується в програмах на різних пристроях. Це також найуживаніша мова програмування у світі.

Як працює Javascript

Щоб знати, як це працює, важливо зрозуміти значення трьох ключових елементів веб-сайту: HTML, CSS і JavaScript.

HTML обробляє весь вміст веб-сайту, включаючи текст, зображення, заголовки сторінок та інший вміст. CSS, з іншого боку, визначає макет вмісту. Це забезпечує розміри зображень і колір шрифту, тип і розмір текстів. Він налаштовує вміст із HTML і охоплює більшість факторів дизайну.

JavaScript оживає всі ці елементи. Це робиться за допомогою анімації зображень, керування мультимедіа (скажімо, відеопрогравачем) та оновлення вмісту. Давайте розглянемо просту інформаційну форму. Якщо у формі запитується вік користувача, користувач може ввести число. Потім це число перевіряється JavaScript, перевіряючи рядки коду (також відомі як «рядки»), чи є воно відповідним значенням. Кнопка для надсилання віку, як-от кнопка «Далі», також працює на основі JavaScript.

Деякі веб-сайти використовують або динамічний код, або статичний код, або обидва. Динамічний означає, що вміст оновлюється залежно від певної обставини, наприклад, поточного курсу іноземної валюти. З іншого боку, статичний код відображає лише вміст, який ніколи не змінюється.

Крім того, JavaScript доповнюється інтерфейсами прикладного програмування (API). Ці попередньо створені блоки коду спрощують передачу даних, дозволяючи людині вбудовувати або реалізовувати програми, які інакше важко кодувати самостійно. Приклади цих API включають:

- API геолокації – контролює географічну інформацію, найбільш очевидну в Google Maps
- topSites API – надає доступ до сайтів, які користувачі найчастіше відвідують
- API Карт Google – вбудовує власні карти та маршрути проїзду для полегшення навігації
- Facebook Login API – дозволяє автентифікацію користувача під час кожного входу

2.1.2 Node.js

Node.js – це надзвичайно потужна платформа на основі JavaScript, яка використовується для розробки онлайн-додатків для чату, сайтів для потокового відео, односторінкових програм та багатьох інших веб-програм і веб-додатків, які інтенсивно використовують введення-виведення. Побудований на движку JavaScript V8 Google Chrome, він використовується як великими, відомими компаніями, так і новоспеченими стартапами (Netflix, Paypal, NASA і Walmart, щоб назвати лише деякі).

Node.js є відкритим і повністю безкоштовним, яким користуються тисячі розробників у всьому світі. Це дає безліч переваг, що робить його кращим вибором, ніж інші серверні платформи, такі як Java або PHP.

Якщо ви новачок у Node.js або тільки ознайомлюєтесь із його основними концепціями, ця стаття дасть вступ до основних основ цієї платформи на основі JavaScript. У наступній статті розповідається про основи цього Node.js, причини його використання, його різні частини, фреймворк Node.js Express, варіанти використання та галузеві тенденції.



Рисунок. 2.2 – Node.js

Поточність

Потоки – це об’єкти, які дозволяють безперервно читати або записувати дані. Існує чотири типи потоків:

1) Читання: це типи потоків, з яких можна зчитувати дані
2) Можливість запису: це типи потоків, у які можна записувати дані
3) Дуплекс: це потоки як для читання, так і для запису
4) Трансформація: потоки, які можуть маніпулювати даними під час їх читання або запису

Домен

Модуль домену перехоплює помилки, які залишаються необробленими. Для перехоплення цих помилок використовуються два методи:

1) Внутрішня прив’язка: випромінювач помилок виконує свій код всередині методу run
2) Зовнішнє прив’язування: випромінювач помилок явно додається до домену за допомогою його методу add

Node.js Express Framework

Express — це гнучкий фреймворк веб-додатків Node.js, який надає широкий набір функцій для розробки веб- та мобільних додатків. Це шар, побудований на вершині Node.js, який допомагає керувати сервером і маршрутами.

Можна виділити основні функції фреймворку Express:

1) Використовується для розробки односторінкових, багатосторінкових і гібридних веб-додатків
2) Дозволяє розробникам налаштувати проміжне програмне забезпечення для відповіді на HTTP-запити
3) Визначає таблицю маршрутизації, яка використовується для виконання різних дій на основі методу HTTP та URL-адреси
4) Дозволяє динамічне відтворення сторінок HTML на основі передачі аргументів шаблонам

NPM: менеджер пакетів вузлів

NPM — одна із найпопулярніших пакетів Node.js. Вона має більше сотні тисяч бібліотек для завантаження, організованих відповідно до конкретних вимог, і є найбільшим реєстром програмного забезпечення у світі. NPM безкоштовний. Ці бібліотеки швидко набирають популярності навіть зараз.

Для чого використовують Node.js?

Із основних способів, які використовують крупні корпорації:

- Бекенд для соціальних мереж
- Додаток для чату
- Передача даних
- Додаток IoT
- Односторінковий додаток

Популярність Node.js

Node.js привернув увагу компаній і організацій з усіх секторів. Це не дивно, враховуючи його універсальність та сильну підтримку громади. Як ви можете бачити з вищезгаданих випадків використання, є досить великі компанії, які використовують Node.js, організації та підприємства, такі як NASA, Uber, PayPal і Netflix.

Нижче наведені основні тенденції галузі, для того щоб краще зрозуміти популярність Node.js

Тенденції галузі

Розробники Node.js користуються попитом у всьому світі завдяки широкому поширенню цієї бібліотеки JavaScript. За версією Forbes, це одна з 10 найбільш затребуваних вакансій.

За даними Node Source, у 2020 році було завантажено 98,9 мільйона бібліотек NodeSource Node.js.

Використання Node.js у виробництві різко зросло з моменту його випуску в 2010 році.

Розробники Node.js отримують кращу заробітну плату, ніж інші розробники веб-технологій. Середня зарплата розробника Node.js в Індії становить 900 000 рупій на рік, а середня зарплата розробника Node.js у Сполучених Штатах становить 115 000 доларів США на рік.

Завдяки таким користувачам, як Netflix, Paypal та інші технологічні компанії, Node.js спостерігає експоненційне збільшення використання у веб-

розробці.

Node.js також певною мірою завдячує своїй популярності своїй асоціації JavaScript. Оскільки JavaScript є найпопулярнішою мовою, як видно з опитування Stack Overflow у 2020 році, багато розробників можуть почати працювати над бібліотекою Node.js без напруженого навчання.

2.1.3 SQLite

SQLite — це реляційна база даних, сумісна з SQL. На відміну від інших систем на базі SQL, таких як MySQL і PostgreSQL, SQLite не використовує архітектуру клієнт-сервер. Вся програма міститься в бібліотеці C, яка вбудовується в програми. База даних стає невід'ємною частиною програми, усуваючи ресурсомісткі автономні процеси.



Рисунок. 2.3 – SQLite

SQLite зберігає свої дані в одному кросплатформному файлі. Оскільки немає виділеного сервера чи спеціалізованої файлової системи, розгортання SQLite так само просте, як підключення його бібліотеки та створення нового

звичайного файлу.

Ця простота призвела до масового прийняття SQLite як системи баз даних для додатків і вбудованих пристроїв. Вважається, що загальна кількість розгортань SQLite перевищує кількість будь-якого іншого механізму баз даних разом узятих, оскільки він постачається з усіма основними операційними системами, більшістю мов програмування, величезним списком вбудованого обладнання та багатьма основними програмними продуктами.

Обмеження SQLite

SQLite пропонує підтримку більшості функцій стандартної мови SQL92. Існують деякі розбіжності та особливості движка, хоча це також справедливо для всіх інших провідних механізмів баз даних SQL. З огляду на це, є деякі специфічні для SQLite обмеження, які варто мати на увазі.

SQLite використовує вільний підхід до обробки типів даних, що суперечить сильному типізації інших механізмів. Він не заперечує проти вставки недійсних значень, тому ви можете написати «sqlite» (рядок) у стовпець цілого числа. Типи даних надзвичайно гнучкі, а іноді й непередбачувані, особливо якщо ви використовуєте іншу систему баз даних.

SQLite також взагалі не підтримує деякі типи даних. Немає BOOLEAN або DATETIME, тому замість них потрібно використовувати текстові або цілі значення. Ці перешкоди можуть викликати головний біль, якщо ви коли-небудь відійдете від SQLite на альтернативну платформу. Ваша база даних може містити недійсні значення SQL, які не будуть прийняті в інших місцях.

Дизайн плоского файлу обмежує варіанти використання, до яких можна

застосувати SQLite. Ви не можете реально масштабувати або розповсюджувати базу даних, оскільки всі підключення записуються в один і той самий базовий файл. Одночасний запис неможливий, оскільки база даних заблокована для кожного. Це знижує продуктивність у сценаріях із важким записом, оскільки операції збираються в чергу на запис.

SQLite також не підтримує кількох користувачів. У механізмах баз даних, таких як MySQL і PostgreSQL, ви можете створювати облікові записи користувачів у базі даних, до яких підключаються клієнти. Це допоможе вам обмежити доступ до певних схем і операцій.

Керування користувачами на рівні бази даних менш застосовне до SQLite, оскільки кожна схема зберігається як звичайний файл на диску. Бази даних SQLite не можуть бути доступні через мережу, тому окремі облікові записи користувачів менш актуальні. Натомість доцільніше використовувати дозволи операційної системи для обмеження дозволів на читання та запис. Тим не менш, відсутність користувачів все ще викликає занепокоєння для програм, що працюють з конфіденційними даними, де кілька ізольованих облікових записів бази даних можуть посилити захист.

Підсумки

SQLite — це SQL-сумісний механізм баз даних, який зберігає все в одному фізичному файлі. Немає сервера для запуску, тому SQLite можна скомпілювати у ваші власні програми. Він безкоштовний, з відкритим вихідним кодом і опублікований у відкритому доступі, тому вам не потрібно турбуватися про ліцензування.

Простота, портативність і надійність двигуна зробили його вибором для

сучасних операційних систем і вбудованих платформ. Він майже не споживає ресурси, є повсюдним і простим у використанні для розробників і працює непомітно для кінцевих користувачів

SQLite виріс на тлі збільшення використання мобільних пристроїв і продуктів Інтернету речей. Прийняття та обізнаність повинні розширюватися, оскільки ці технології продовжують розширювати своє охоплення. Ключем до успіху SQLite була його універсальна сумісність — якщо у вас є бібліотека SQLite і деяке сховище, ви можете використовувати потужність SQL без будь-яких інших залежностей.

2.1.4 Heroku

Heroku — це платформа як послуга (PaaS), яка дозволяє розробникам створювати, запускати й керувати додатками повністю в хмарі.
Ця платформа пропонує підтримку різноманітних мов програмування.
Heroku запускає програми через віртуальні контейнери, відомі як Dynos.

Простими словами, Heroku — це платформа, яка використовується для розгортання та запуску сучасних програм.

Офіційно підтримувані мови в Heroku

Heroku офіційно підтримує наступні мови програмування:

- Java
- Ruby
- PHP
- Node.js
- Python

- Scala
- Clojure



Рисунок. 2.4 – Heroku

Чому Heroku?

Heroku дуже зручний для розробників. Це досить хороший початок, якщо мова йде про новачків, які хочуть навчитися розгортанню програмного забезпечення та хостингу. Для початківців розробників Heroku безкоштовний. Це добре для розміщення підтвердження концепцій та особистих проектів. Безкоштовний рівень пропонує:

- Розгортання за допомогою Git і Docker
- Кастомні домени
- Контейнерна оркестровка
- Автоматичне встановлення патчів ОС

Heroku надає всю підтримку розробникам, командам і підприємствам будь-якого розміру, щоб розгортати керування та легко масштабувати свої програми.

Heroku надає чудові плагіни та підтримку сторонніх розробників. Крім офіційно підтримуваних мов, Heroku пропонує підтримку будь-якої мови, яка працює в Linux, за допомогою стороннього пакета збірки. Вони також надають понад 200 додаткових послуг, які можна легко інтегрувати за кілька кліків.

Крім цього, Heroku володіє всіма загальними та очікуваними перевагами хмарних обчислень. Коли програми масштабуються, як розробники, ми не хочемо керувати серверами. Heroku орієнтована на розробників і пропонує нам зосередитися на кодуванні та дозволяє Heroku керувати серверами для використання в хмарі.

Якщо говорити про те, що він орієнтований на розробників, у Heroku також є потужний CLI. Ви можете просто завантажити та встановити його на будь-яку платформу ОС. За допомогою Heroku CLI ви можете легко створювати програми та керувати ними безпосередньо через термінал. Це важлива частина використання Heroku.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА СТВОРЕННЯ ТЕЛЕГРАМ БОТА

3.1 Створення телеграм бота

Для того, щоб створити телеграм бота будемо використовувати офіційного помічника-бота **@botfather** . Спочатку потрібно найти його в мережі telegram.



Рисунок. 3.1 – телеграм-бот BotFather

Використовуючи команди **/newbot** створюємо нового бота. Після чого пропонується ввести назву чат-бота з обов'язковою умовою: в кінці має бути **Bot** або **_bot** . Якщо ж всі умови для створення були виконано, то **BotFather** видає спеціальний набір символів для доступу HTTP API Telegram Bot і URL-адресу для доступу до чат-боту.

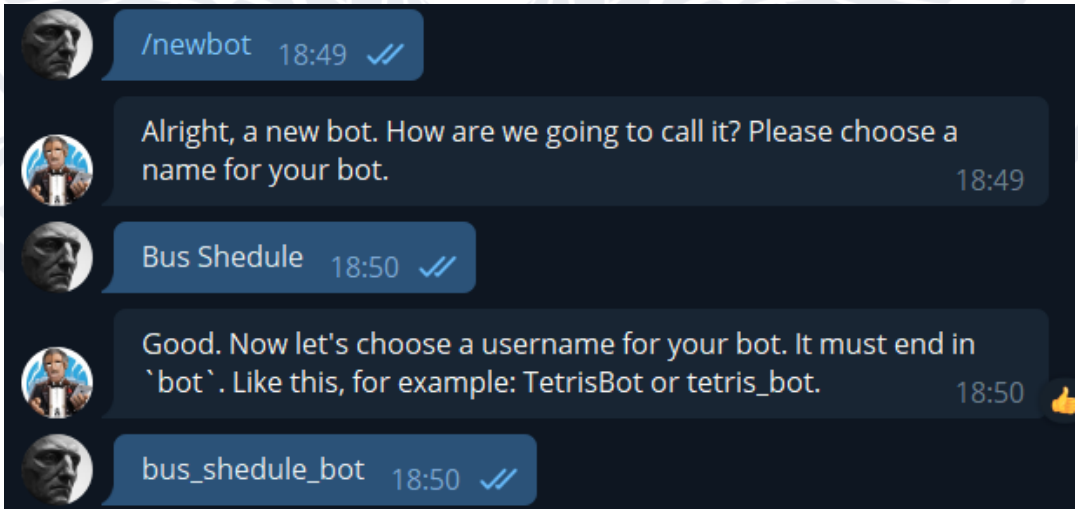


Рисунок. 3.2 – процес створення телеграм-боту

Для установки додаткових параметрів, таких як іконка чат-бота, вітальне повідомлення, опис чат-бота, а так само видалення наявних чатботи, існують такі команди:

- /setname – змінює ім’я бота
- /setdescription - змінює опис бота
- /setabouttext – змінює інформацію про бота
- /setuserpic – змінює фотографію профіля бота
- /setcommands – змінює список команд
- /deletebot – видаляє бота

Використовуючи команду **/setabouttext** змінюємо опис боту

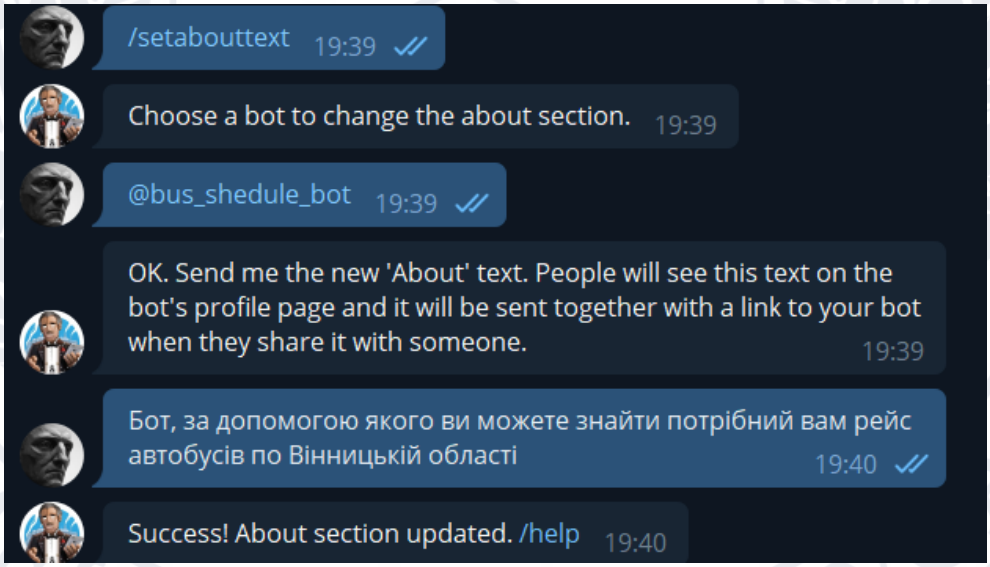


Рисунок. 3.3 – зміна опису бота

Далі, потрібно змінити аватар телеграм-боту, потрібно використати команду **/setuserpic**

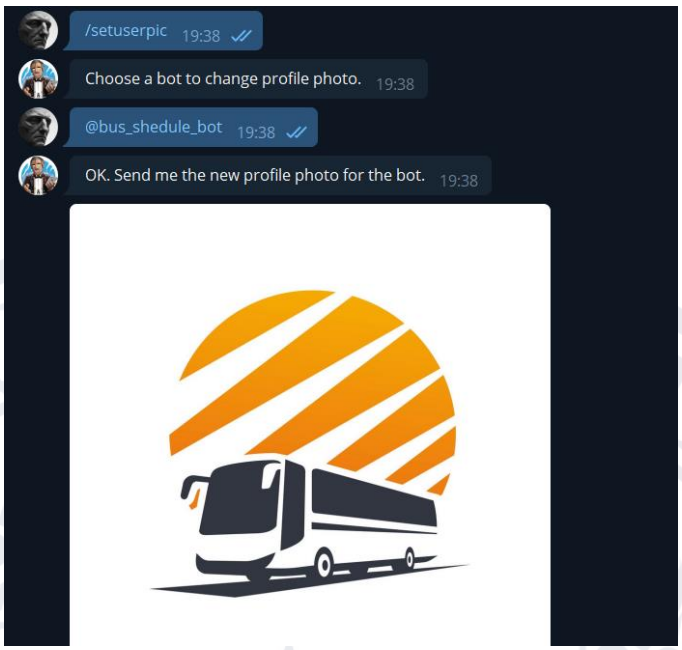


Рисунок. 3.4 – зміна фото бота

Після виконаних операцій, наш бот тепер має підходящий зовнішній вигляд:

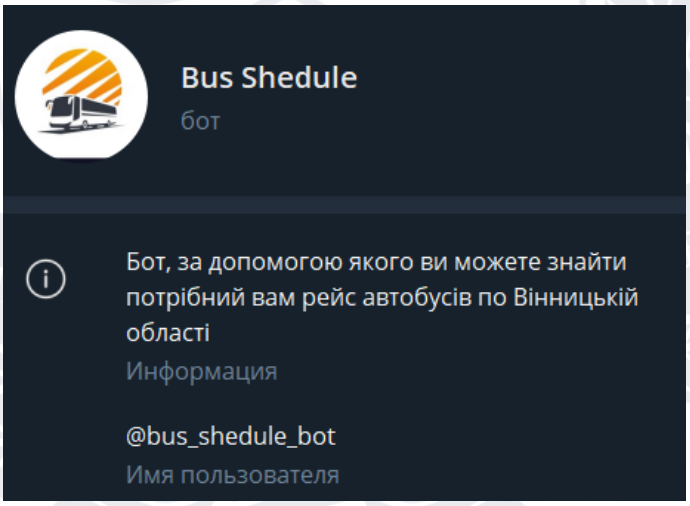


Рисунок. 3.5 – вигляд @Bus_shedule_bot

У месенджері Телеграм кожна клавіатура реалізована як об’єкт,а її кнопки відповідно як формат для зберігання і обміну отриманої інформації, яку користувач отримує для читання.

Документація Telegram Bot API передбачає, що кожна клавіатура повинна мати один обов'язковий параметр – назва кнопки, та шість необов'язкових – посилання, зворотній зв'язок, можливість вбудованого запиту, можливість виведення клавіатури з інших чатів, виклик опису запущеної гри та кнопка зможливістю покупки.

У розробленому @bus_shedule_bot були використані наступні бібліотеки:



Рисунок. 3.6 – використані бібліотеки

- **MomentJS** — це бібліотека JavaScript, яка допомагає дуже легко аналізувати, перевіряти, маніпулювати та відображати дату/час у JavaScript. Moment JS дозволяє відображати дату відповідно до локалізації та в зрозумілому для користувача форматі.
- **sequelize -** інструмент ORM node.js, який забезпечує просту роботу з базою даних надаючи абстракції для роботи sql, робить роботу більше безпечною так як захищає від SQL Injection
- **sqlite3 -** база даних, яка забезпечує просте зберігання інформації та дозволяє отримати доступ до бази даних за допомогою запитів sql

- **telegraf** - бібліотека, яка спрощує розробку власних ботів Telegram за допомогою JavaScript
- **nodemon** - інструмент, за допомогою якого можна розробляти програми на основі node.js шляхом автоматичного перезапуску програми, коли виявлено зміни файлів у коді.

3.2 Розробка телеграм бота

Спочатку створюємо папку з проектом, потім відкриваємо консоль і

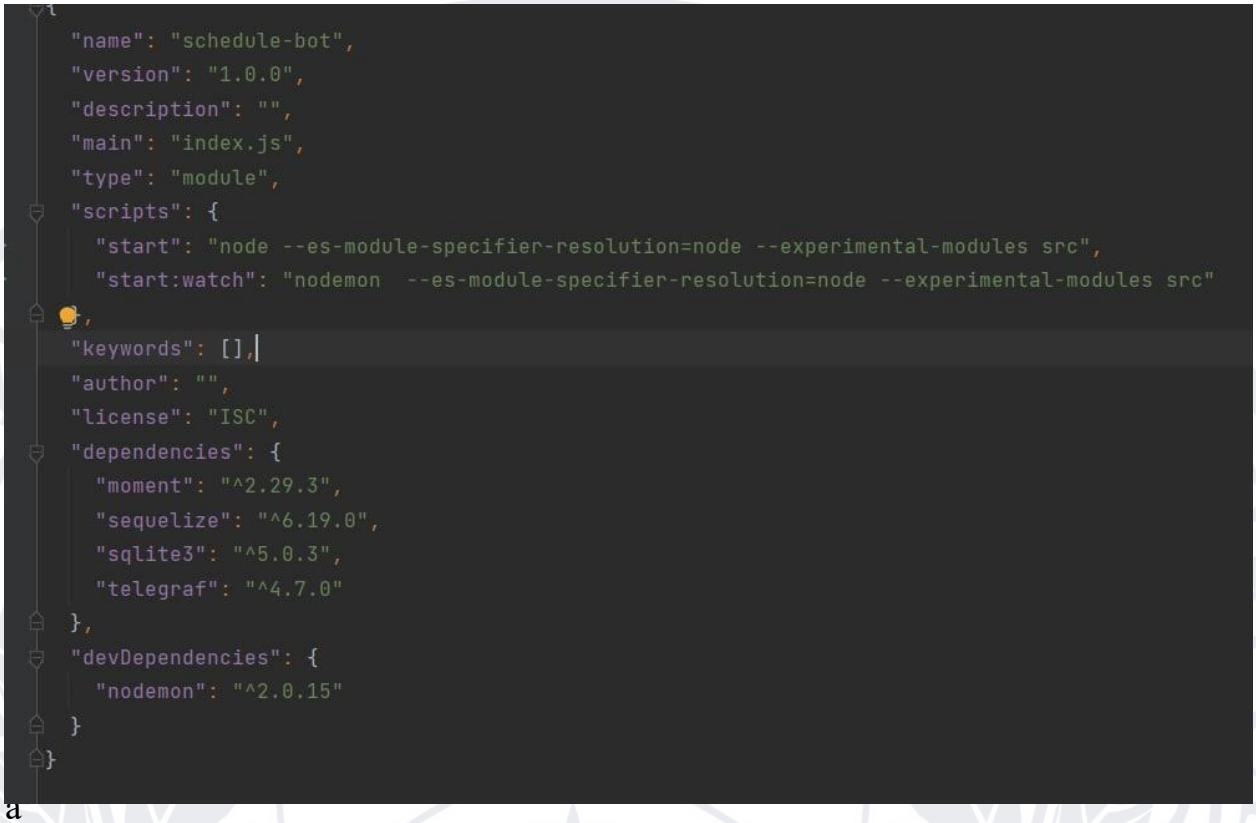


Рисунок. 3.7 – створення папки з проектом

Після чого встановлюються залежності за допомогою команди npm розпочати розробку коду.

Спочатку потрібно розробити структуру бази даних: звя’зки, назви колонок, назви таблиць. Після того як було продумано вигляд бази даних, потрібно описати структуру бази в коді.

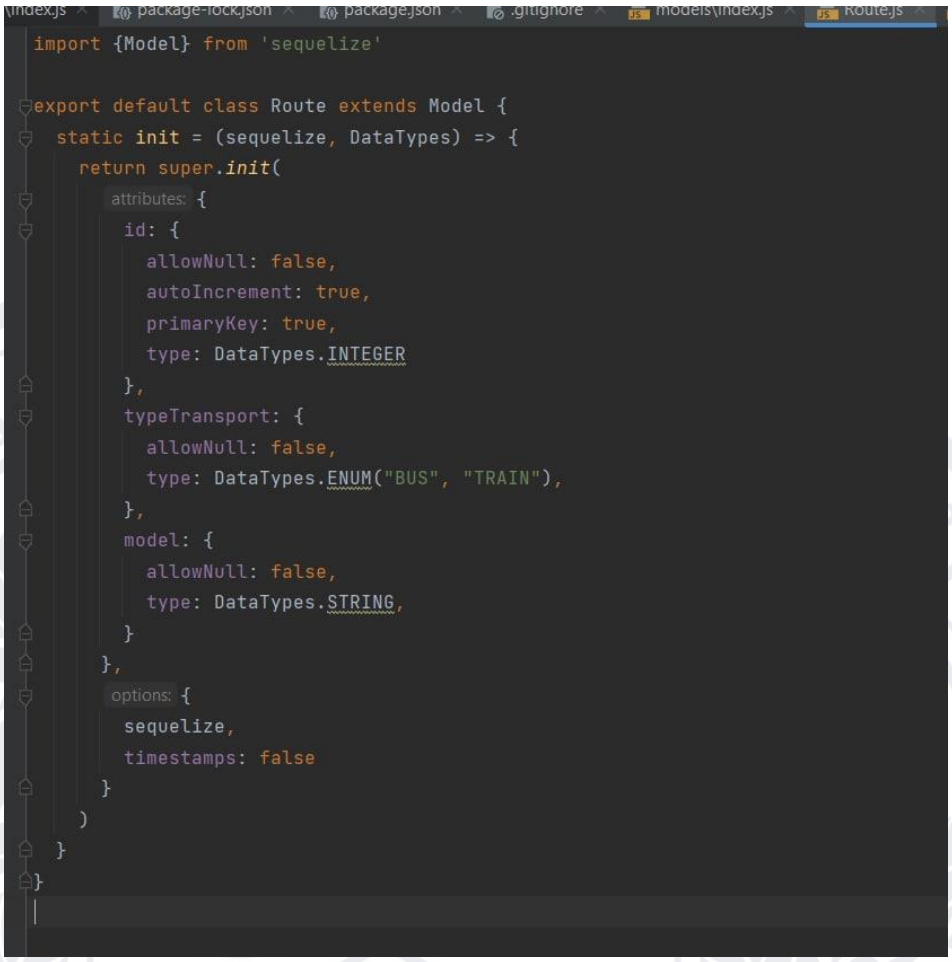


Рисунок. 3.8 – структура бази даних

Для цього створюємо папку models в папці src. В папці models ми описуємо дані моделі

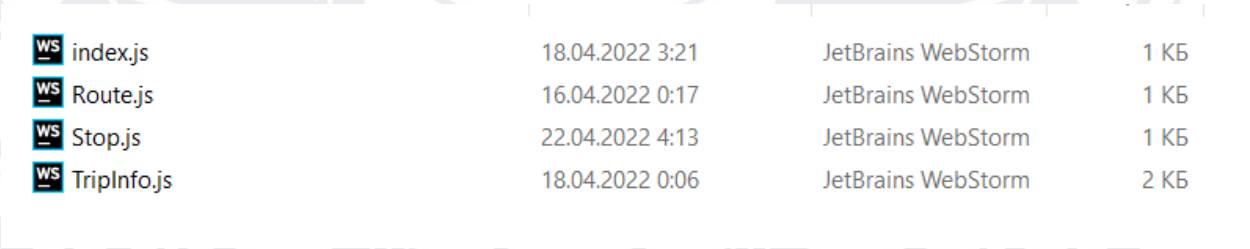


Рисунок. 3.9 – опис данних

Ось так виглядає описана модель:

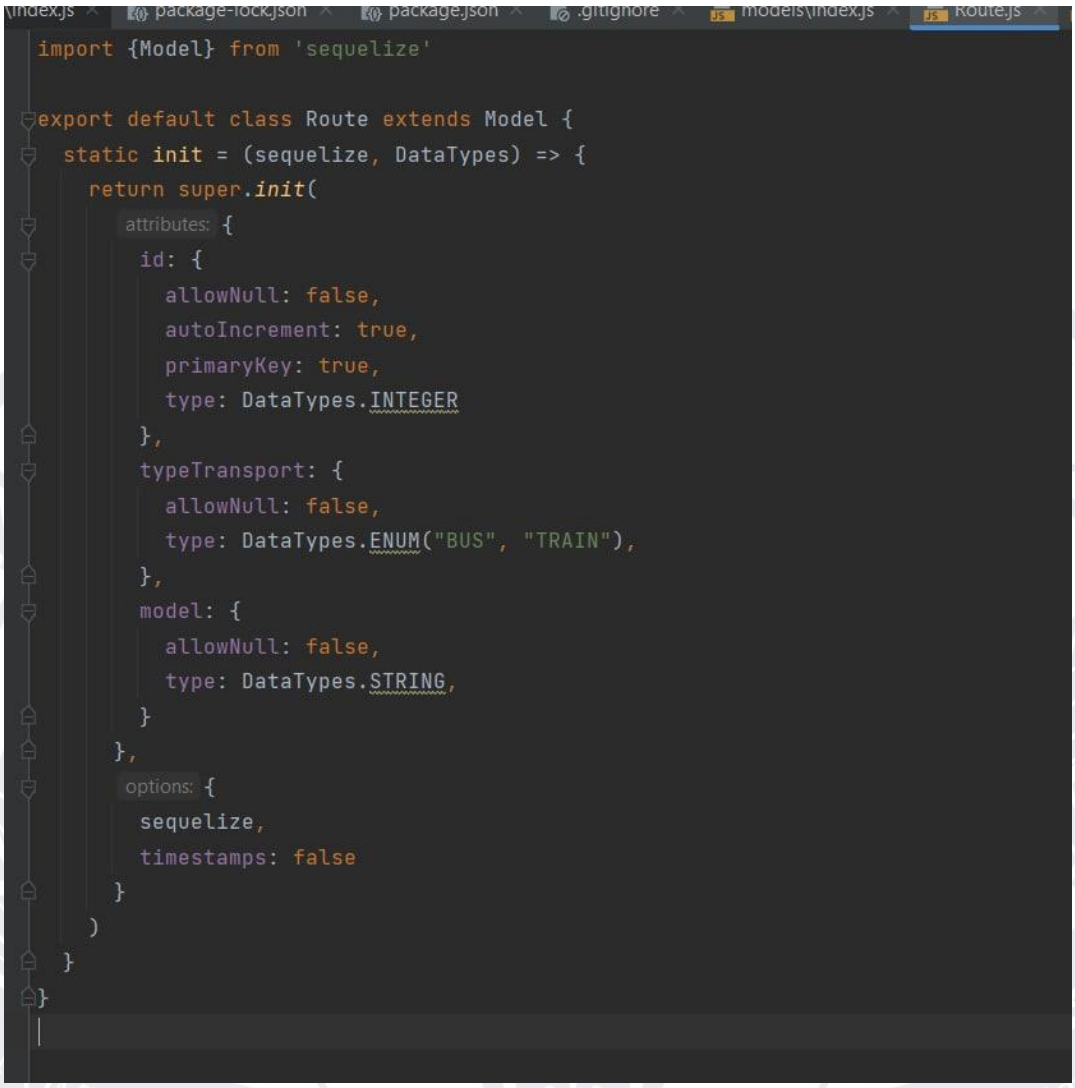


Рисунок. 3.10 – опис моделі

І аналогічно описуємо інші моделі, як можна побачити на зображені було описано структурні поля бази даних, була вказана структура і тип поля

Після того як були створені 3 моделі, потрібно їх підключити. Нижче відбувається підключення sequelize до бази даних і з’єднання моделей з sequelize

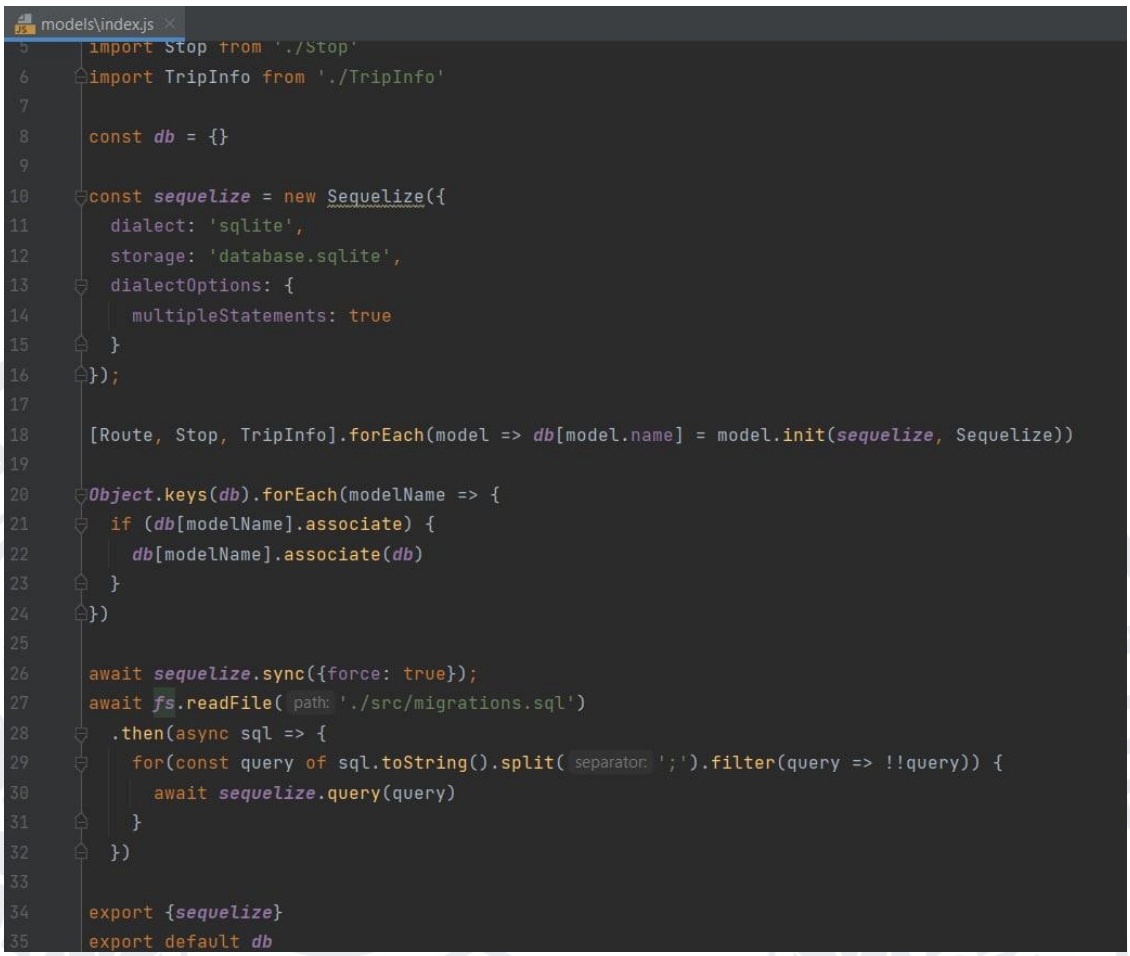


Рисунок. 3.11 – підключення до бази даних

Далі показано як було реалізоване підключення до мережі Telegram і написана логіка реакції телеграм-боту на дії користувача.

На 9 стрічці коду (рис. 3.12) можна побачити підписку на подію початку запуску боту користувачем. На 10 стрічці задається функція, яка буде реагувати на введення тексту користувачем і відповідно логіка описана в функції *handleTextMessage.*

В даній функції відбуваються наступні дії:

- перевірка на коректність назви населеного пункту
- перевірка того, що точка відправлення не співпадає з точкою прибуття
- відбувається запит на отримання інформації про маршрути без

пересадок

- отриманий результат форматується до заданого стилю після чого відправляється користувачеві
- отримання маршрутів з пересадками (форматування і відправлення)



Рисунок. 3.12 – підключення до Telegram

В функціях отримання маршрутів основна логіка, вибірка потрібних маршрутів відбувається за допомогою sql. За допомогою sql були зроблені перевірки на логіку часу (щоб час прибуття першого автобусу був раніше чим час відправлення другого).



```
        f."order"     as fromTransferOrder,
        t."order"     as toTransferOrder,
        ff.*,
        tt.*
from (
         select *
         from TripInfos
         where routeId in (:fromRouteIds)
     ) f
         join (
    select *
    from TripInfos
    where routeId in (:toRouteIds)
) t on f.stopId = t.stopId
         join (select routeId, name, "order"
               from TripInfos
                        JOIN Stops S on S.id = TripInfos.stopId
               WHERE name = :from) ff on ff.routeId = f.routeId
         join (select routeId, name, "order"
               from TripInfos
                        JOIN Stops S on S.id = TripInfos.stopId
               WHERE name = :to) tt on tt.routeId = t.routeId
where fromTransferOrder > ff."order"
  and tt."order" > toTransferOrder
  and f.arrivalTime < t.departureTime
GROUP BY f.routeId, t.routeId;
```

Рисунок. 3.13 – перевірка логіки часу

Далі розглянемо функцію форматування результатів пошуку вказаних користувачем маршрутів.

Як можна побачити на рисунку нижче форматується зручне для перегляду повідомлення:

- назва маршрутів
- час відправлення
- час прибуття
- тип транспорту
- назва транспорту
- виділення синім кольором точку відправлення та відповідно точку прибуття, для того, щоб користувачу було зручніше переглядати інформацію



Рисунок. 3.14 – реалізація пошуку маршрутів

Далі в файлі migration.sql потрібно задаються дані для роботи системи:

- назви зупинок
- список маршрутів
- інформація про рейси

migrations.sql

```sql
INSERT INTO Stops (name)
VALUES ('Вінниця');

INSERT INTO Stops (name)
VALUES ('Гайсин');

INSERT INTO Stops (name)
VALUES ('Тульчин');

INSERT INTO Stops (name)
VALUES ('Стрижавка');

INSERT INTO Stops (name)
VALUES ('Піщанка');

INSERT INTO Stops (name)
VALUES ('Жмеринка');

INSERT INTO Stops (name)
VALUES ('Козятин');

INSERT INTO Stops (name)
VALUES ('Ладижин');

INSERT INTO Stops (name)
VALUES ('Хмільник');

INSERT INTO Routes (typeTransport, model)
VALUES ('BUS', 'MA3-232');
```

Рисунок. 3.15 – дані для роботи системи

Далі реалізована команда для запуску телеграм-боту.

Після того, як вся логіка роботи була розписана створюємо функцію запуску телеграм-боту і викликаємо її.

index.js

```js
import Tg from './services/Tg'

Tg.launch().then(() => console.log('Bot started'))
```

Tg.js

```js
import {Telegraf} from 'telegraf'
import {TELEGRAM_TOKEN} from "../config";
import BuilderTrip from "./BuilderTrip";

class Tg {
  constructor() {
    this.bot = new Telegraf(TELEGRAM_TOKEN)

    this.bot.start(this.handleStart)
    this.bot.on( updateType: 'text', this.handleTextMessage)
  }

  launch() {
    return this.bot.launch()
  }
```

Рисунок. 3.16 – команда запуску бота

3.3 Логіка роботи телеграм-боту

Для того, щоб розпочати роботу з телеграм-ботом потрібно завантажити додаток Telegram . Версію для персональних комп'ютерів можна завантажити з сайту https://desktop.telegram.org/?setln=uk .

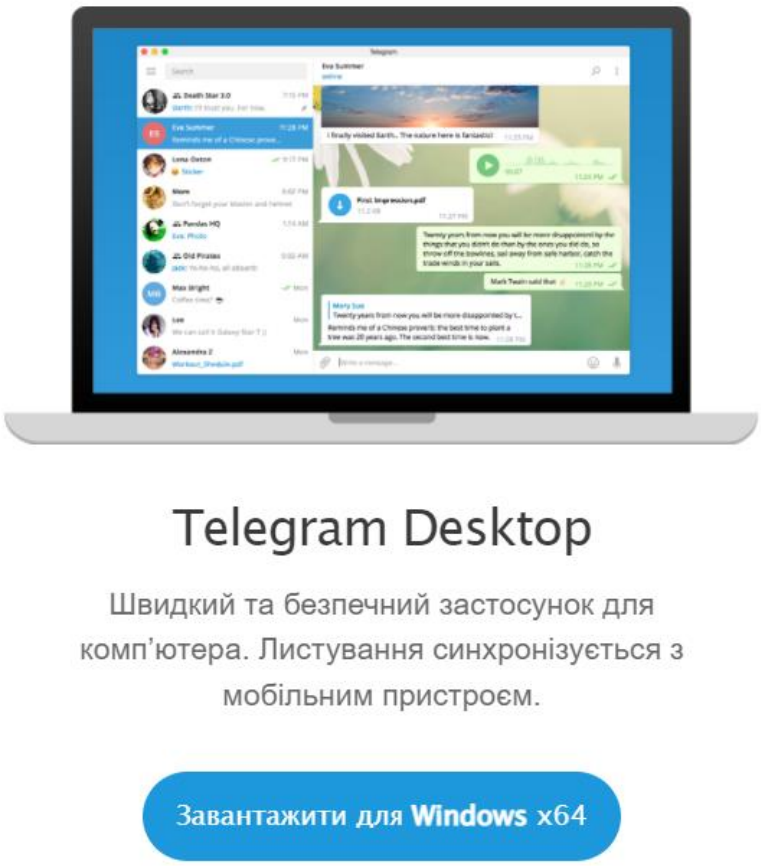


Рисунок. 3.16 – додаток Telegram

Для користувачів мобільних девайсів з операційною системою Android потрібно відкрити PlayMarket та для власників IOS необхідно відкрити AppStore та ввести в пошуку "Telegram" відповідно завантаживши його.

Далі потрібно ввести номер телефону, який прив'язаний до аккаунта, або ж якшо аккаунт відсутній виконати швидку реєстрацію.

Після того як вхід в Telegram був виконаний в строці пошуку треба ввести **@bus_shedule_bot**. Клікаємо на бота, після чого можемо ним користуватись



Рисунок. 3.17 – телеграм-бот bus_shedule_bot

Для зручності пошуку телеграм-бота можна використати QR- код:

Далі необхідно натиснути команду /start , після чого бот готовий виконувати пошук потрібних користувачу маршрутів автобусів по Вінницькій області.

Ось так виглядає початок взаємодії телеграм-бота з користувачем:

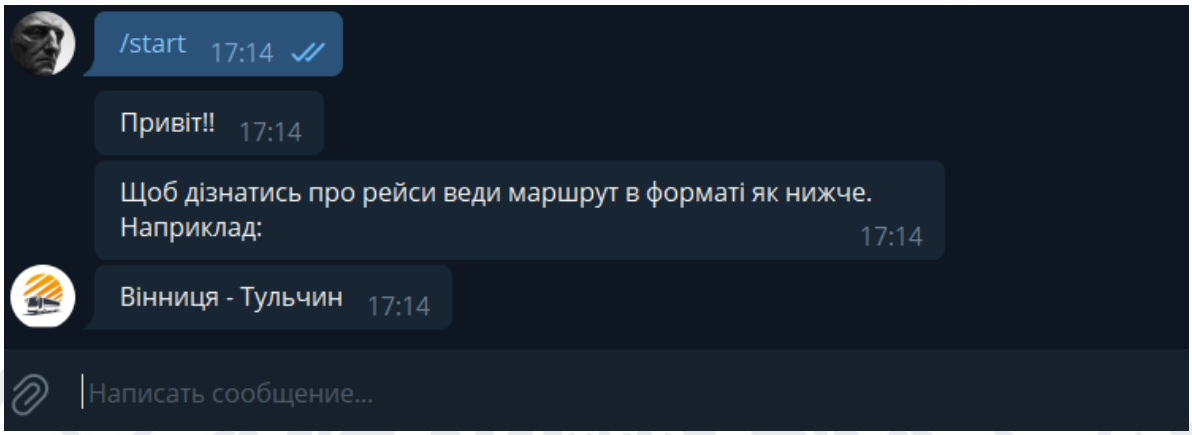


Рисунок. 3.18 – взаємодія користувача з ботом

Користувач може виконати пошук потрібного йому маршруту написавши рейс в наступному форматі: (Пункт відправлення) - (Пункт прибуття).

Доступні для телеграм-бота містя для пошуку: Вінниця, Гайсин, Тульчин, Стрижавка, Піщанка, Жмеринка, Козятин, Ладижин, Хмільник

Введемо для прикладу наступний пошук маршруту:

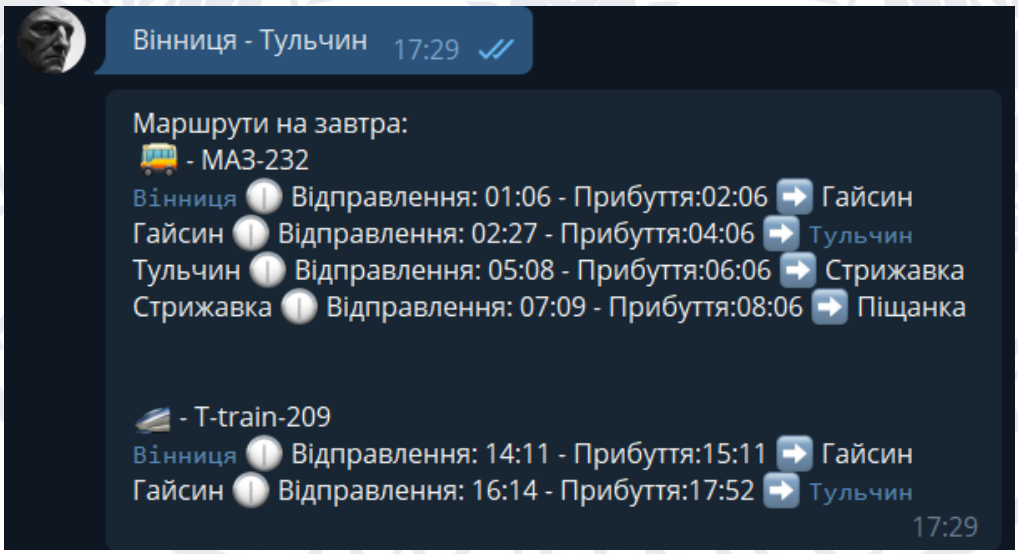


Рисунок. 3.19 – інформація про рейс

У випадку якщо прямого рейсу не знайдено телеграм-бот покаже користувачу маршрут з пересадкою. Наприклад:

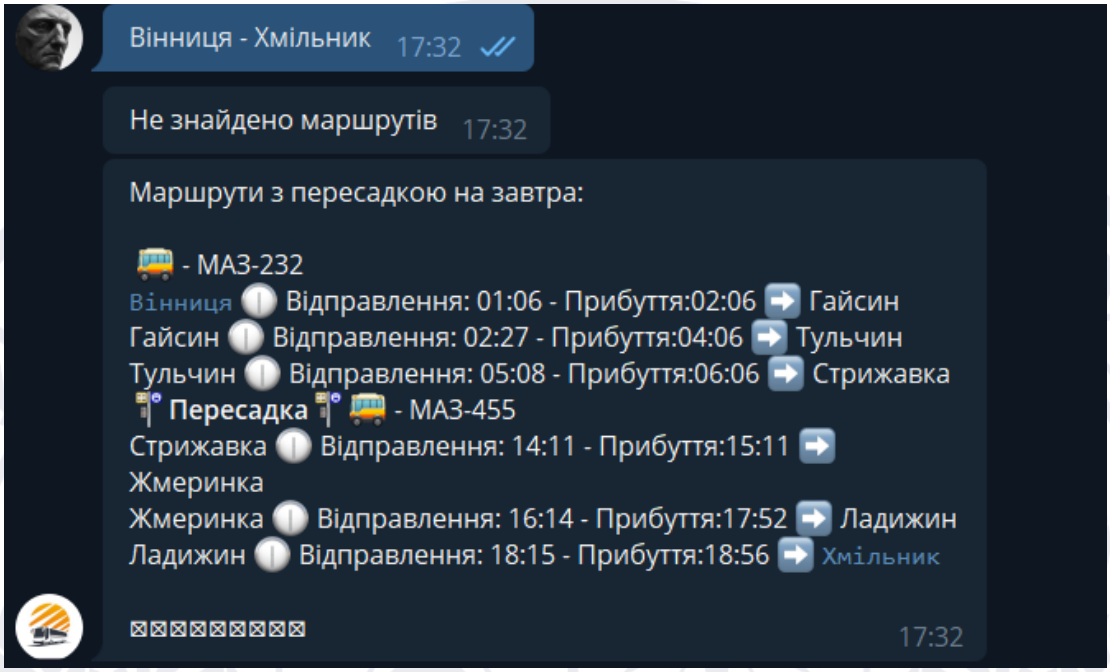


Рисунок. 3.20 – маршрут з пересадкою

Якщо ж виконаний пошук маршруту, який відсутній в базі даних, то користувач отримає наступне повідомлення:

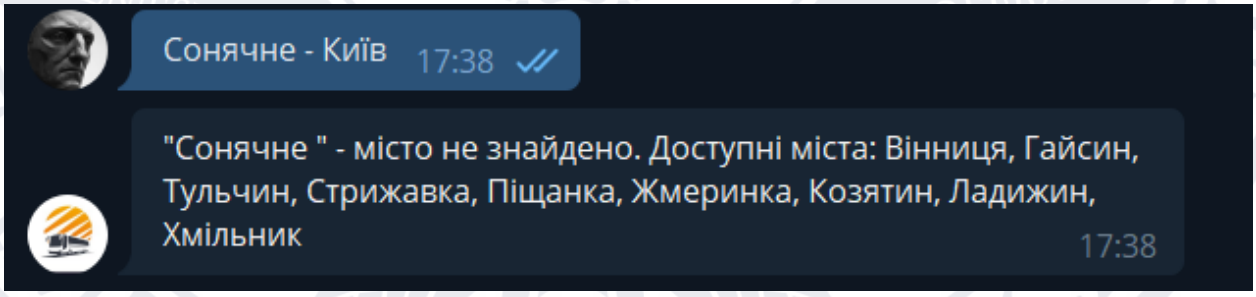


Рисунок. 3.21 – відсутній маршрут

ВИСНОВОК

У бакалаврській роботі представлені результати, які відповідають поставленій меті та є вирішенням задачі проектування додатку для керуванняфінансами з функцією візуалізації даних.

Для вирішення задачі були проаналізовані такі аналоги:

- AirTrack Ua
- RailwayBot
- Depovnua_bot

Для вирішення проблеми були вибрані найсучасніші технології, тому що в них актуальна документація і реалізовані всі базові можливості що дозволяє зосередитись на написанні логіки.

Для розробки був використаний JavaScript так як ще одна з найпопулярніших мов і вже написано багато бібліотек. Telegraf ми використовували щоб було простіше взаємодіяти з телеграмом.

Для збереження даних про маршрути ми використовували Sqlite це маленька легковісна база даних, як не вимагає складної установки і може працювати навіть в памяті.

Основна функція це пошук маршруту, також пошук маршруту з пересадкою. Всі дані виводяться в достатньо зручному форматі тобто відображається час, тип траспорту, міста через які прямую даний машрут. Надалі було би доцільно сконцентруватись на додаванні нових міст і маршрутів, плюс це можна було би додати інші види транспортів наприклад авіарейси, також ще можна додати ціни за квиток і кількість вільних місць

СПИСОК ВИКОРИСТАНИХ ПОСИЛАНЬ

1. JavaScript [Електронний ресурс] –Режим доступу до ресурсу: https://blog.hubspot.com/website/what-is-javascript
2. Npm-init [Електронний ресурс] – Режим доступу до ресурсу: https://docs.npmjs.com/cli/v8/commands/npm-initJavaScript
3. Telegraf.js [Електронний ресурс] –Режим доступу до ресурсу:https://telegraf.js.org/#:~:text=Telegraf%20is%20a%20library%20that,bots%20using%20JavaScript%20or%20TypeScript.
4. Node.js [Електронний ресурс] – Режим доступу до ресурсу: https://ua.wikipedia.org/wiki/Node.js
5. Heroku [Електронний ресурс] – Режим доступу до ресурсу:https://www.heroku.com/about
6. Heroku [Електронний ресурс] – Режим доступу до ресурсу: https://levelup.gitconnected.com/all-you-need-to-know-about-heroku-in-5-minutes-7d4ec8849114
7. SQLite [Електронний ресурс] – Режим доступу до ресурсу: https://www.sqlite.org/about.html#:~:text=SQLite%20is%20an%20in-process,any%20purpose%2C%20commercial%20or%20private.
8. SQLite [Електронний ресурс] – Режим доступу до ресурсу: https://www.sqlitetutorial.net
9. SQLite [Електронний ресурс] – Режим доступу до ресурсу:https://www.howtogeek.com/devops/what-is-sqlite-and-why-is-it-so-popular/

10. Heroku [Електронний ресурс] – Режим доступу до ресурсу: https://mentormate.com/blog/what-is-heroku-used-for-cloud-development/
11. Telegraf [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/telegraf/telegraf
12. WebStorm [Електронний ресурс] – Режим доступу до ресурсу: https://www.jetbrains.com/ru-ru/webstorm/
13. Nodemon [Електронний ресурс] – Режим доступу до ресурсу: https://www.digitalocean.com/community/tutorials/workflow-nodemon-ru
14. Sequelize [Електронний ресурс] – Режим доступу до ресурсу: https://habr.com/ru/post/565062/
15. Telegram на Python. Heroku сервер [Електронний ресурс]. – Режим доступу: https://tproger.ru/translations/telegram-bot-create-and-deploy/
16. Telegram-bot[Електронний ресурс].– Режим доступу: https://habr.com/ru/post/442800/
17. Рейтинг мессенджерів[Електронний ресурс].– Режим доступу: https://coba.tools/compilation/reiting-messendzherov-2018
18. Telegram bot API[Електронний ресурс] – Режим доступу до

ресурсу: https://tlgrm.ru/docs/bots/api#authorizing-your-bot

19. Топ 30 телеграм-ботів[Електронний ресурс] – Режим доступу до ресурсу: https://ain.ua/ru/2017/08/14/30-telegram-botov/
20. Федотенко М.А. Розробка мобільних додатків, перші кроки. 2019. С. 140-144
21. Bot API 2.0: информация для разработчиков. Режим доступу: https://tlgrm.ru/docs/bots/2-0-intro
22. Telegram bot: що це таке і як ним користуватися. Режим доступу: https://kyiv.dityvmisti.ua/pages/telegram-bot-shcho-tse-take-i-iak-nymkorystuvatysia.
23. Дослідження українського телеграм: хто, чомуі як використовує мессенджер. Режим доступу:https://vctr.media/ukrainskiytelegram-5000-15716/
24. Приховані можливості Telegram про які ви, можливо, не знали. Режим доступу: https://blog.comfy.ua/ua/prikhovani-mozhlivosti-telegram-proyaki-vi-mozhlivo-ne-znali/
25. Функції Telegram, про які ти не знав. Режим доступу: https://studway.com.ua/pro-telegram/

26. Бот в Телеграм. Режим доступу: https://lemarbet.com/ua/razvitie-internetmagazina/boty-v-telegram/

27. Офіційний сайт Telegram. Режим доступу: https://telegram.org

28. Telegram. Режим доступу: https://uk.wikipedia.org/wiki/Telegram

29. Роботи та інформація для розробників. Режим доступу: https://tlgrm.ru/docs/bots

Декларація щодо унікальності текстів роботи
та невикористання матеріалів інших авторів без посилань

Стецюк Сергій Олегович
Прізвище, ім’я, по батькові
Факультет інформаційних і прикладних технологій
Факультет
122 Комп’ютерні науки
Шифр і назва спеціальності
Сучасні інформаційні технології та програмування
Освітня програма

ДЕКЛАРАЦІЯ

Усвідомлюючи свою відповідальність за надання неправдивої інформації, стверджую, що подана кваліфікаційна (бакалаврська) робота на тему: «Розробка телеграм-боту для пошуку міжміських автобусних рейсів Вінницької області» є написаною мною особисто.

Одночасно заявляю, що ця робота:

- не передавалась іншим особам і подається до захисту вперше;
- не порушує авторських та суміжних прав, закріплених статтями 21-25 Закону України «Про авторське право та суміжні права»;
- не отримувались іншими особами, а також дані та інформація не отримувались у недозволений спосіб.

Я усвідомлюю,. що у разі порушення цього порядку моя кваліфікаційна (бакалаврська) робота буде відхилена без права її захисту, або під час захисту за неї буде поставлена оцінка «незадовільно».

_____ _____
дата підпис