

Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»

Навчально-науковий  
інститут електроенергетики

(інститут)

Факультет інформаційних технологій

(факультет)

Кафедра інформаційних технологій та комп'ютерної інженерії  
(повна назва)

**ПОЯСНЮВАЛЬНА ЗАПИСКА**  
**кваліфікаційної роботи ступеня магістра**

студента Волкова Владислава Руслановича  
(ПІБ)

академічної групи 123М-22-1  
(шифр)

спеціальності 123 «Комп'ютерна інженерія»  
(код і назва спеціальності)

за освітньо-професійною програмою 123 «Комп'ютерна інженерія»  
(офіційна назва)

на тему «Обґрунтування структури комплексу системи комплексу кінотеатрів  
«Планета кіно» міста Дніпро з урахуванням бота продажі квитків»  
(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	проф. Цвіркун			
розділів:				
теоретичний розділ	проф. Цвіркун Л.І.			
синтез системи	доц.Бешта Д.О.			
розроблення програмного забезпечення	ас. Панферова Я.В.			

Рецензент				
-----------	--	--	--	--

Нормоконтролер	доц. Шедловська Я.І			
----------------	---------------------	--	--	--

Дніпро  
2023

**ЗАТВЕРДЖЕНО:**  
завідувач кафедри  
інформаційних технологій  
та комп'ютерної інженерії  
(повна назва)

\_\_\_\_\_ Гнатушенко В.В.  
(підпис) (прізвище, ініціали)

"06" вересня 2023 року

**ЗАВДАННЯ**  
**на кваліфікаційну роботу**  
**ступеня магістра**

студента Волков В.Р. академічної групи 123М–22–1  
(прізвище та ініціали) (шифр)

спеціальності 123 «Комп'ютерна інженерія»  
(код і назва спеціальності)

за освітньою–професійною програмою 123 «Комп'ютерна інженерія»  
(офіційна назва)

на тему «Обґрунтування структури комплексу системи комплексу кінотеатрів  
«Планета кіно» міста Дніпро з урахуванням бота продажі квитків»

затверджену наказом ректора НТУ «Дніпровська політехніка» від 09.10.2023 № 1227

Розділ	Зміст	Термін виконання
Стан питання та постановка завдання	На основі матеріалів виробничих практик, інших науково–технічних джерел конкретизується предмет та мету роботи та виконується постановка завдання	10.10.2023
Теоретичний	Обґрунтувати теоретичну базу розв'язання наукового завдання, якому присвячено роботу	25.10.2023
Синтез системи	Розробка комп'ютерної системи	15.11.2023
Розроблення програмного забезпечення	Розробка програмного забезпечення	29.11.2023
Експериментальний розділ	Проведення і обробка результатів експериментів	06.12.2023

**Завдання видано**

\_\_\_\_\_ (підпис керівника)

проф. Цвіркун Л.І.

(прізвище, ініціали)

**Дата видачі**

06 вересня 2023

**Дата подання до екзаменаційної комісії**

10.12.2023

**Прийнято до виконання**

\_\_\_\_\_ (підпис керівника)

Волков В.Р.

(прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка 74 с., 24 рис., 1 дод., 10 джерел

TELEGRAM, БОТ, PYTHON, API, ТЕЛЕГРАМ-БОТ

Об'єкт розробки: комплекс системи кінотеатрів «Планета кіно» у місті Дніпро.

Мета роботи: розробка Telegram-бота для продажі квитків для мережі кінотеатрів «Планета кіно» міста Дніпро та обґрунтування вибору цієї структури.

Пояснювальна записка має аналіз сучасних систем кінотеатрів та їх інтеграції з ботом для продажу квитків.

За допомогою отриманих даних було сформовано завдання дослідження.

У теоретичному розділі проведено наукове обґрунтування та побудовано модель системи кінотеатрів «Планета кіно» з урахуванням бота продажі квитків. Вирішено наукове завдання побудови ефективної структури комплексу.

У розділі «Синтезу системи» сформульовані технічні вимоги до створення системи, враховано особливості сфери кінотеатрів та взаємодії з ботом для продажу кіно. Побудовано структурну схему та функціональну схему.

У розділі «Розроблення програмного забезпечення» проведено розробку програмного забезпечення, базуючись на побудованих схемах алгоритмів. Описано зв'язок між програмними та їх функціональні можливості, зокрема, взаємодію з ботом для продажу квитків.

В експериментальному розділі, проведено експеримент, використовуючи створену систему з ботом для продажі квитків. Аналізовано результати та виявлено переваги нової структури.

## ЗМІСТ

Перелік умовних позначень, символів, скорочень і термінів.....	6
Вступ.....	7
1 Основна частина.....	11
1.1 Стан питання.....	11
1.2 Критичний аналіз і класифікація напрямків досліджень у сфері IoT технологій і телеграм-ботів з використанням API в комп'ютерних мережах..	11
1.3 Ідея впровадження telegram-бота та інтеграція IoT-пристроїв в мережу кінотеатрів «планета кіно».....	13
1.4 Постановка задачі.....	14
2 Теоретична частина.....	15
2.1 Характеристика і структура об'єкта впровадження.....	15
2.1.1 Стислі відомості про мережу впровадження системи обліку для кінотеатрів «планета кіно».....	20
2.2 API та актуальність технології і її вживаність.....	22
2.3 Телеграм боти та їх застосування.....	23
2.4 Використання телеграм-ботів у поєднанні з API технологією.....	23
2.5 IoT технології та їх використання.....	24
3 Розділи синтезу системи продажі квитків.....	26
3.1 Вибір і обґрунтування принципів побудови функціональної схеми проєктованого об'єкту.....	26
3.2 Формулювання технічних вимог до системи продажі квитків.....	27
3.2.1 Вимоги до реалізації системи.....	27
3.2.2 Вимоги до функцій системи.....	27
3.2.3 Вимоги до видів забезпечення.....	27
3.2.4 Вимоги до захисту інформації.....	28
3.2.5 Вимоги до ергономіки системи.....	28
3.3 Обґрунтування прийнятих способів проєктування і дослідження.....	28
3.4 Синтез структури за заданими показниками системи.....	30
3.5 Проєктування системи взаємодії телеграм – бот та системи оброблювання квитків.....	31
3.5.1 Розробка функціональних схем вузлів телеграм – бота проєктованої системи купівлі квитків.....	32
3.5.2 Вибір апаратних засобів і елементної бази.....	36
4 Розроблення програмного забезпечення.....	38
4.1 Призначення й область застосування програмного забезпечення.....	38
4.2 Проєктування принципових схем вузлів і блоків.....	38
4.3.1 Загальні відомості.....	47
4.3.2 Функціональне призначення.....	47
4.3.3 Опис логічної структури програми.....	47

5	Експериментальний розділ.....	50
	5.1 Сутність експерименту (мета, умова) .....	50
	5.2 Результат експерименту в цифрах і фактах.....	53
	5.3 Аналіз відповідності теоретичних та експериментальних досліджень... ..	56
	5.4 Характеристика новизни результатів.....	59
	Висновок .....	61
	Перелік посилань.....	63
	Додаток А. Текст програми розробки Telegram-бота.....	67

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

IoT – Internet of Things

API – Application programming interface

SOAP – Simple Object Access Protocol

ПЗ – Програмне забезпечення

DHCP – Dynamic Host Configuration Protocol

IP – Internet Protocol

NAT – Network Address Translation

RPC – Remote procedure call

REST – Representational state transfer

## ВСТУП

На даний час, у зв'язку зі стрімким розвитком інформаційних технологій, все більшої актуальності набуває напрямок впровадження інтелектуальних пристроїв та Інтернету речей. Враховуючи глобалізацію інформатизації суспільства в області інформаційних систем, можна з упевненістю сказати, що майбутнє без пристроїв, підключених до Інтернету, здається неможливим. Разом з тим, однією з основних тенденцій розвитку інформаційних систем та технологій стає розширення доступності інформаційно-обчислювальних ресурсів мереж для окремих користувачів.

Пристрої Інтернету речей використовуються не лише у щоденному житті, а й у сучасному бізнес-середовищі, що супроводжується активним поширенням їх у різноманітних галузях та сферах, внаслідок чого пристрої IoT стають невід'ємною частиною нашого життя.

У наш час, повсякденне життя людей тісно пов'язане з використанням комп'ютерних технологій, і це призводить до зростаючої популярності соціальних мереж та месенджерів. Один із найбільш популярних месенджерів – це Telegram, завдяки своїм зручностям, простоті та надійності у використанні, а також сучасним налаштуванням для безпечного спілкування. Саме ці характеристики роблять його вибором багатьох користувачів.

Один з ключових елементів Telegram – це чат-боти, які дозволяють автоматизувати рутинні процеси. Ця тема є дуже актуальною і користується великим попитом, оскільки люди проводять значний час в Інтернеті, зокрема в соціальних мережах і месенджерах. Зазвичай, користувачам зручніше спілкуватися через повідомлення, ніж за допомогою дзвінків, і це пояснює поширену популярність чат-ботів. Чат-боти дозволяють спростити комунікацію між користувачами та організацією, а також забезпечують доступ до необхідної інформації 24/7.

Виходячи з цього розробка та реалізація IoT технологій та телеграм-бота для мережі кінотеатрів «Планета кіно» є одним із способів полегшити та покращити взаємодії між відвідувачами та співробітниками кінотеатру. У зв'язку з цим прийнято рішення здійснити налаштування контролю температури в приміщенні та забезпечити протипожежну безпеку, за допомогою «Інтернет речей» та упровадити системи контролю до комп'ютерного комплексу кінотеатрів «Планета кіно» міста Дніпро. Додатково створити апаратно – програмну систему контролю вторгнення нелегітимних користувачів та запобігання стороннього доступу до кінозалів, та використані технології API у телеграм ботах для одержання інформації про послуги в кінотеатрі та одержання повідомлень про порушення або навпаки про проходження автентифікації користувачів, які забезпечуються працездатність кінцевого апаратно – програмного комплексу кінцевого рішення. Дана комп'ютерна система являє сукупність, фізичної частини розробленого апаратного рішення, з кодом програми застосунку. Сучасні мережеві рішення спираються на програмно–апаратний комплекс, що забезпечує взаємодію співробітників підприємства.

**Мета та завдання дослідження.** *Метою роботи є обґрунтування структури комплексу системи кінотеатрів «Планета кіно» в місті Дніпро з урахуванням впровадження бота для продажу квитків. У процесі дослідження необхідно розглянути такі завдання:*

- охарактеризувати стан розвитку системи комплексу кінотеатрів «Планета кіно» міста Дніпро;
- проаналізувати вимоги до структури комплексу з урахуванням бота продажі квитків;
- розробити Telegram-бота для продажі квитків, який буде відповідати всім поставленим вимогам.

*Об'єкт дослідження* – корпоративна мережа комплексу кінотеатрів «Планета кіно» міста Дніпро.



*Предмет дослідження* є структура та організація комплексної системи кінотеатрів «Планета кіно», з фокусом на впровадження бота для продажу квитків та його вплив на споживачів.

*Методи дослідження* – для досягнення поставленої мети виконано аналіз літературних джерел та публікацій, що стосуються побудови корпоративних мереж, засобів безпеки, розробки телеграм-ботів мовою програмування Python. Та слід використати такі методи для досягнення поставленої мети:

- аналітичний метод для вивчення стану розвитку системи комплексу кінотеатрів «Планета кіно» міста Дніпро;
- метод експертних оцінок для визначення вимог до структури комплексу системи кінотеатрів «Планета кіно» з урахуванням бота продажі квитків;
- метод системного аналізу для розробки структури комплексу системи.

*Наукові положення:*

1. Структура комплексу системи кінотеатрів «Планета кіно» буде ґрунтуватися на таких наукових положеннях:

- система управління кінотеатром є сукупністю взаємодії взаємопов'язаних підсистем, які забезпечують функціонування кінотеатру ;
- структура комплексу системи управління кінотеатром повинна відповідати його функціональним вимогам та особливостям;
- впровадження бота продажі квитків до комплексу системи управління кінотеатром дозволяє підвищити ефективність його роботи;
- розробка та програмування телеграм-ботів для обміну даними;

*Наукові результати:*

1. У результаті дослідження розроблена модель бота продажі квитків для комплексу системи кінотеатрів «Планета кіно» міста Дніпро. Модель бота забезпечує всі необхідні функції та відповідає сучасним вимогам до системи автоматизації продажів.

**Обґрунтованість та достовірність наукових положень, висновків та рекомендацій** базується на аналізі надійних джерел і врахуванні практичного досвіду з розробки Telegram-бота мовою програмування Python.

**Практичне значення отриманих результатів** полягає в розробці Telegram-бота для комплексу кінотеатрів «Планета кіно», для підвищення конкурентоспроможності та ефективності роботи кінотеатру, та для покращення взаємодії користувачів з співробітниками кінотеатру.

## ОСНОВНА ЧАСТИНА

### 1.1 Стан питання

Кіноіндустрія важлива складова сфери культури. Кіно допомагає особистості відгородитися від проблем, зануритися в іншу реальність. У 21 столітті кіно – не спосіб розважитися, а джерело для початку мислення, аналізу, а також пізнання. Кіноіндустрія – потужний важіль для розвитку країни.

Мережа кінотеатрів «Планета кіно» є частиною розвитку духовного та культурного просвітлення і розважальної індустрії. Таким чином в цій галузі дуже важливу роль займає своєчасне впровадження нових технологій для заохочення нових клієнтів та підвищення їх безпеки та зручності вживання послухякі може надати кінотеатр. Проектоване програмне технічне рішення допоможе такому бізнесу та користувачам створювати та споживати послуги у більш зручному та інформативному форматі. Тому потрібно старити комплекс контролю з використання IoT – пристроїв та створити програмне забезпечення для телеграм – бота так і для розумних пристроїв контролю.

### 1.2 Критичний аналіз і класифікація напрямків досліджень у сфері IoT технологій і телеграм-ботів з використанням API в комп'ютерних мережах

Розробка комп'ютерних мережі з використанням IoT технологій та розробки телеграм-ботів з використанням API в наш час є досить актуальною задачею, оскільки ці два напрямки мають великий потенціал для інновацій і застосувань в різних сферах життя та бізнесу. Критичний аналіз може визначити, які з цих напрямків найбільш важливі або потребують найбільшої уваги з погляду інновацій та розвитку.

Напрямки досліджень у сфері IoT технологій можна класифікувати:

– безпека IoT – цей напрям класифікується на вивченні загроз та кібербезпеки, пов'язані з IoT пристроями та розробляє заходи для їх запобігання,

а також включає в себе аналіз вразливостей, вдосконалення протоколів безпеки та створення системи моніторингу та виявлення вторгнення для IoT;

- мережа та зв'язок IoT – цей напрям фокусується на розробці, оптимізації та удосконаленні мережевих технологій та протоколів, що використовуються для зв'язку між підключеними пристроями IoT;

- застосування IoT у різних галузях – IoT технології застосовуються в різних галузях, наприклад в сільському господарстві, в медицині, логістиці, освіті, енергетиці та багато іншого;

- збір та аналіз даних IoT – цей напрям є одним з ключових напрямків цієї технології, оскільки велика кількість даних, зібраних з різних датчиків та пристроїв, надає можливість для розумного прийняття рішень і виявлення цінних зв'язків;

- інтеграція та стандартизація відіграють важливу роль в розвитку та розширенні IoT речей. Інтеграція забезпечує спільну роботу різних пристроїв, систем і платформ, щоб створити цілісний екосистемний підхід для вирішення певних задач, а стандартизація відіграє важливу роль у визначенні загальних правил та норм, які дозволяють різним виробникам та розробникам створювати сумісні та безпечні IoT рішення;

- розробка апаратного забезпечення та датчиків є ключовою частиною розвитку IoT речей, оскільки допомагає реалізувати потенціал у різних галузях;

- аналіз впливу та бізнес моделі є важливим аспектом для розуміння та визначення ефективної стратегії впровадження IoT в підприємствах.

Дослідження у сфері телеграм-ботів з використанням API методу класифікують за кількома напрямками і виходячи з цього можна провести критичний аналіз кожного з них:

- технічний аспект: сюди відносять використання API Telegram (досліджуються технічні аспекти API Telegram, а саме методи API, автентифікація, обмеження і можливості API), мови програмування та інструменти (оцінювання мови програмування та інструментів, які найкраще

підходять для створення телеграм-бота), оптимізація продуктивності (досліджується оптимізація продуктивності телеграм-ботів через API, включаючи швидкість обробки запитів та оптимізацію ресурсів);

– сфери застосування: сюди відносять бізнес рішення (аналіз Telegram-бота в сфері бізнесу для автоматизації клієнтського обслуговування), освіта та навчання (досліджується використання телеграм-ботів для навчання та допомоги студентам), медицина та охорона здоров'я (досліджується, як телеграм-боти використовуються в медичній сфері);

– питання безпеки та конфіденційності: сюди відносять захист даних (аналіз питання захисту даних та конфіденційності в контексті телеграм-ботів), вразливості та атаки (дослідження вразливостей та можливі атаки на телеграм-боти через API та розробка заходів безпеки);

– інтерфейс та користувацький досвід: дизайн та взаємодія (розробка інтерфейсу Telegram-бота більш зрозумілим та зручним), персоналізація (дослідження можливостей персоналізації взаємодії з телеграм-ботами).

Отже, інновації в області IoT технологій та розробки телеграм-ботів з використанням методу API відкривають нові можливості для покращення різних сфер життя та бізнесу. Критичний аналіз цих напрямків дозволив визначити найважливіші аспекти та напрями подальших досліджень.

### **1.3 Ідея впровадження Telegram-бота та інтеграція IoT-пристроїв в мережу кінотеатрів «Планета кіно»**

Впровадження Telegram-бота та IoT-пристроїв в мережу кінотеатрів «Планета кіно» може бути корисним для покращення зручності та ефективності обслуговування клієнтів та оптимізації системи.

Проаналізувавши попередній пункт цієї роботи, можна виділити такі ідеї:

– впровадження API в поєднанні з телеграм ботом для підтримки клієнтів, а саме надання відповідей на питання глядачів, надавати інформацію про адресу та контактну інформацію кінотеатру;

– здійснення контролю доступу до залів кінотеатру використовуючи IoT систему.

#### **1.4 Постановка задачі**

Метою дипломного проекту є обґрунтування структури системи комплексу кінотеатрів «Планета кіно» міста Дніпро з урахуванням бота продажі квитків.

Для досягнення цієї мети необхідно розглянути та вирішити такі питання:

- проаналізувати існуючу систем комплексу кінотеатрів, зокрема, систему продажі квитків на основі Telegram-бота;
- провести аналітичний метод для дослідження сучасних підходів до автоматизації процесів продажі квитків у кінотеатр;
- провести експериментальне дослідження ефективності роботи нової структури;
- розробити програмне забезпечення бота продажі квитків;
- оформити проект у вигляді дипломної роботи.

У сучасних умовах розвитку інформаційних технологій зростає роль автоматизації бізнес-процесів. Це стосується і сфери розваг, зокрема, кінотеатрів. Автоматизація продажі квитків дозволяє поліпшити якість обслуговування клієнтів, а також підвищити ефективність роботи кінотеатру.

Впровадження бота продажі квитків є перспективним інструментом автоматизації, який дозволяє значно спростити та прискорити процес покупки квитків для клієнтів кінотеатрів. Бот може працювати 24/7, дозволяє клієнтам переглядати афішу з фільмами, дізнатись про розклад сеансів, отримати допомогу від працівників кінотеатру, під час сенсу, дізнаватись актуальні новини про кінотеатр, отримати відповіді на питання, самостійно купувати квитки на сеанси.

Бота продажі квитків, має бути написаний мовою програмування Python, з використанням API. Ця система дозволить поліпшити якість обслуговування клієнтів і підвищити ефективність роботи кінотеатру.

## 2 ТЕОРЕТИЧНА ЧАСТИНА

### 2.1 Характеристика і структура об'єкта впровадження

Компанія «Планета кіно» має два кінотеатри в місті Дніпро. Перший знаходиться в ТРК «ARPOLO» за адресою вулиця Титова, 36, Дніпро, Дніпропетровська область, 49000, а другий в ТРК «МОСТ-сіті» за адресою улиця Глінки, 2, Дніпро, Дніпропетровська область, 49000. Відстань між двома офісами складає 5.03 км. Топологічна схема розміщення кінотеатрів представлена на рисунках 2.1 – 2.2.

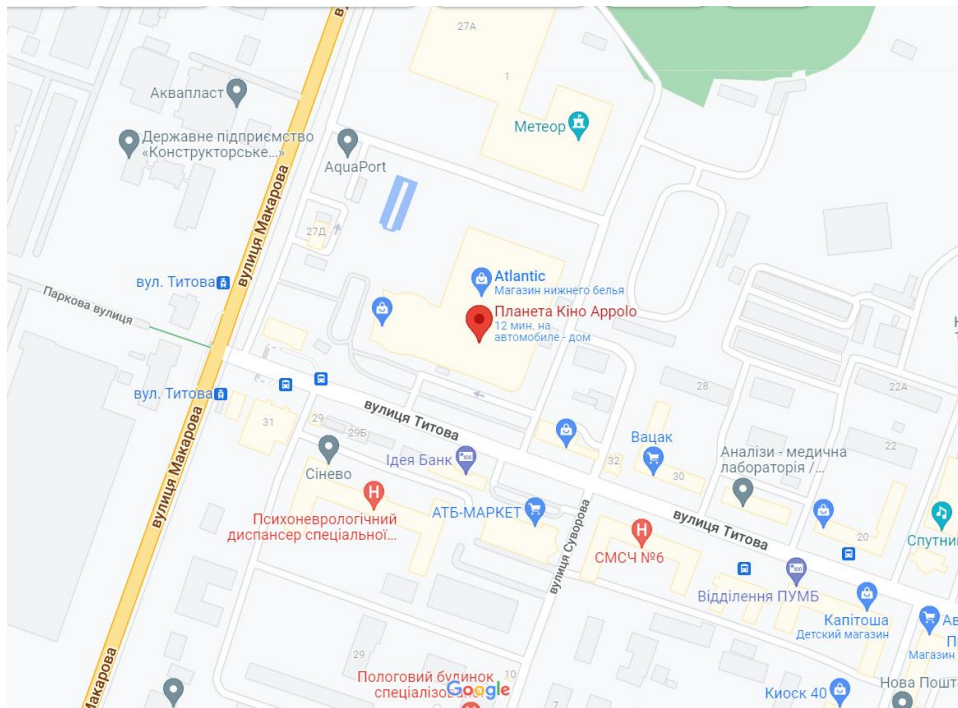


Рисунок 2.1 – Топологічна схема розміщення кінотеатру в ТРК «ARPOLO»

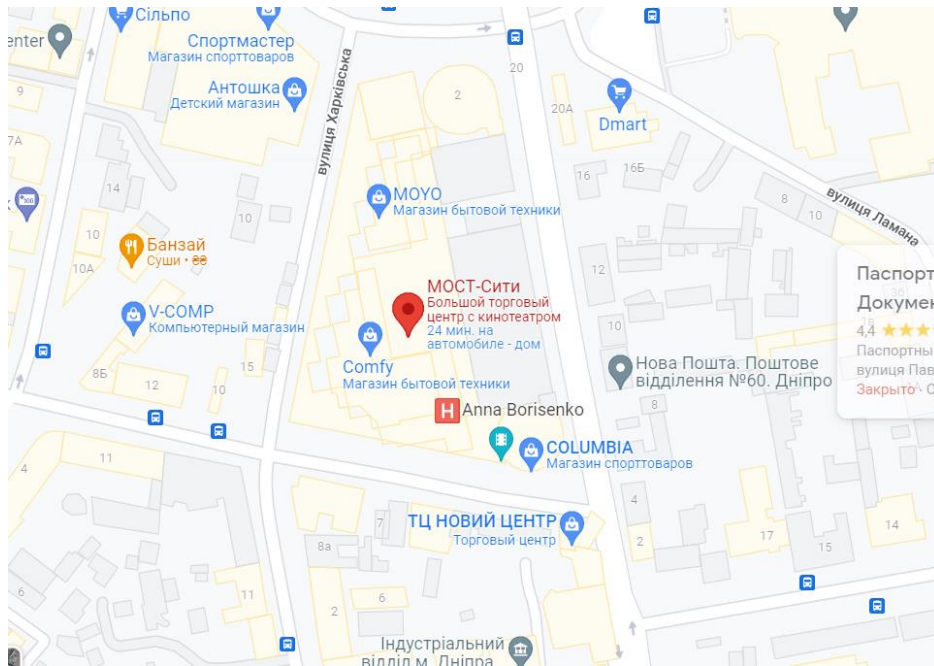


Рисунок 2.2 – Топологічна схема розміщення кінотеатру в ТРК «МОСТ–сіті»

Організаційна структура мережі кінотеатрів «Планета кіно» складається з наступних підрозділів:

- адміністративний відділ;
- фінансовий відділ;
- відділ управління персоналом;
- відділ маркетингу та продажу;
- відділ технічного забезпечення.

Адміністративна частина: керівник підприємства та підлеглі йому заступник та помічники. Керівник повинен мати доступ до усіх систем та надавати всю інформацію, стосовно робочого процесу до головного офісу.

Фінансовий відділ: бухгалтер та помічник. Бухгалтерія повинна мати можливість отримувати звіт від відділу продаж.

Відділ управління персоналом: керівник відділу кадрів, HR–менеджер. HR–менеджер виконує роботу з персоналом особисто та звітує безпосередньо перед головним керівником.

До відділу маркетингу та продаж входять: головний менеджер та його



персонал. Головний менеджер отримує звіти з продаж від кожного свого працівника та звітує його перед головним керівником та бухгалтерією.

Відділ технічного забезпечення: Головний інженер, кіномеханік та ІТ-спеціаліст, забезпечують надійне функціонування комп'ютерної та іншої техніки, її розвиток та адаптацію, автоматизацію різних напрямків діяльності мережі кінотеатру.

Офісна частина в ТРК «APPOLO» займає два поверхи та має такі відділи: на першому поверсі відділ маркетингу та продажу, відділ технічного забезпечення, фінансовий відділ, на другому поверсі адміністративний відділ.

Офісна частина в ТРК «МОСТ-сіті» має такі відділи: відділ управління персоналом.

Організаційна структурна схема наведена в рисунку 2.3.

План приміщення наведено на рисунках 2.4 – 2.5.

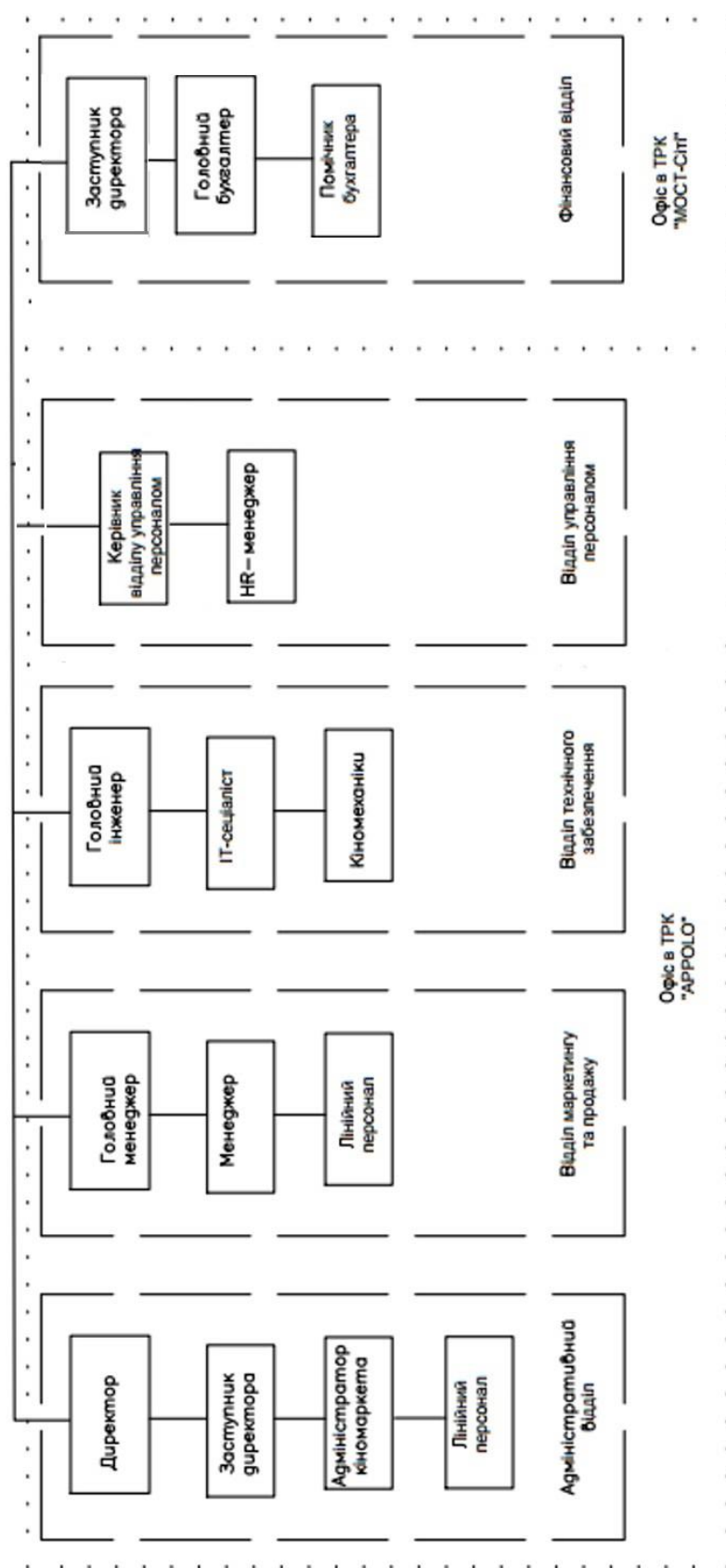


Рисунок 2.3 – Організаційна структурна схема

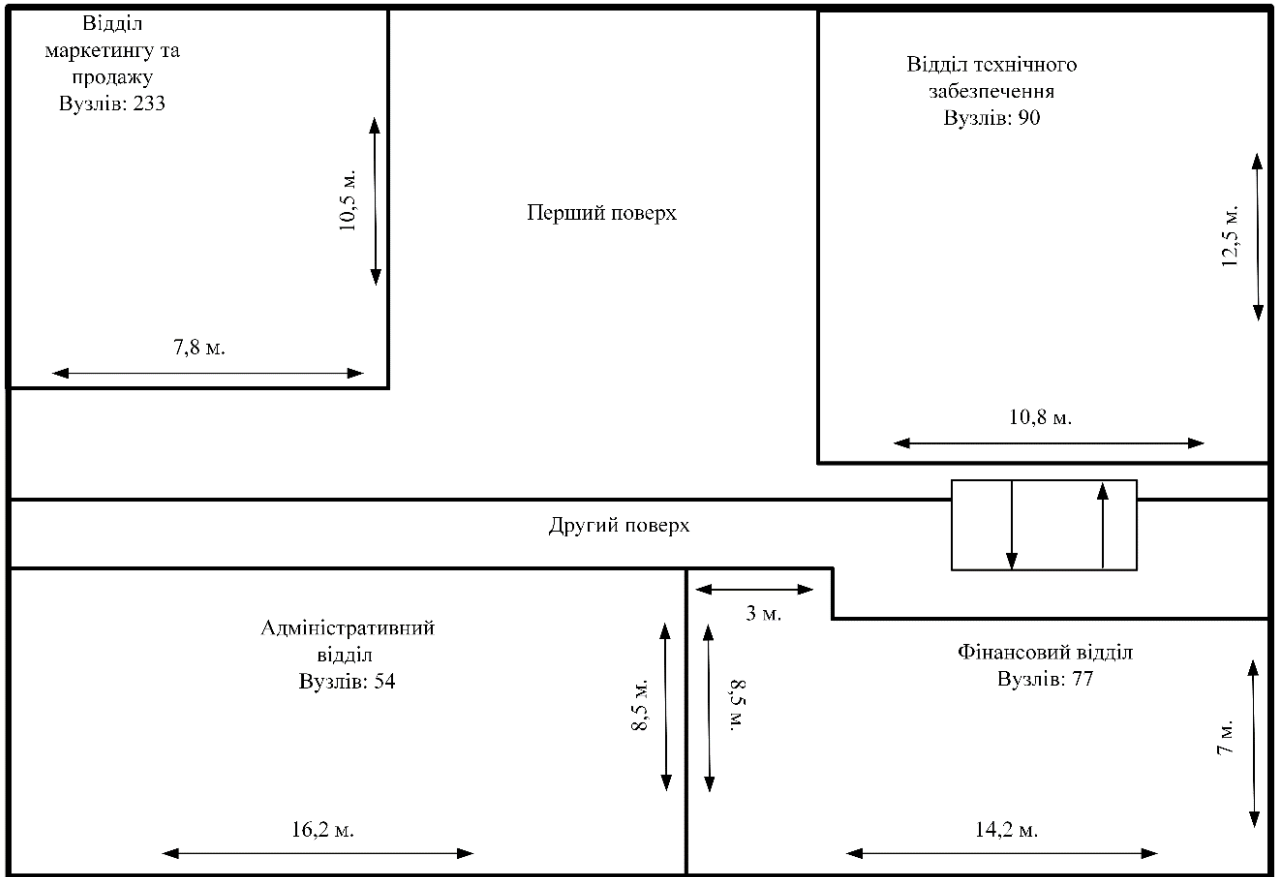


Рисунок 2.4 – План офісного приміщення в ТРЦ «ARPOLO»

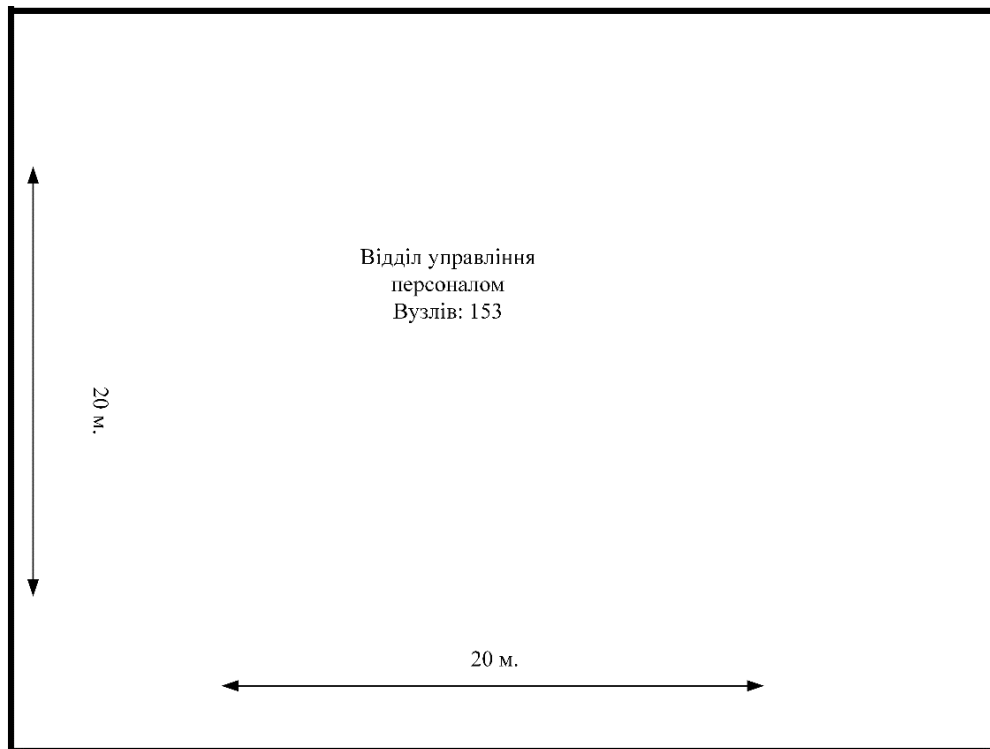


Рисунок 2.5 – План офісного приміщення в ТРЦ «МОСТ–сіті»

### **2.1.1 Стислі відомості про мережу впровадження системи обліку для кінотеатрів «Планета кіно»**

Однією із вимог замовника для побудови мережі було використання обладнання Cisco. Головні переваги даного обладнання:

- забезпечення високого рівня адаптивності бізнесу рахунок автоматизації мережі;
- автоматичне відновлення даних та їх резервування після можливих збоїв;
- відмінні показники параметрів надійності та відмовостійкості;
- виняткова продуктивність бездротових мереж із високою щільністю підключення;
- комплексний підхід до контролю нормальної працездатності мереж та докладне інструктування щодо усунення збоїв;
- мінімізація вразливості у будь-яких точках мережі;
- можливість застосування аналітики для оптимізації продуктивності, програмного забезпечення;

Для рішення поставлених завдань використано сервіс DHCP, що дозволяє комп'ютерам автоматично отримувати IP-адресу та інші параметри, необхідні для роботи в мережі та обрано протокол динамічної маршрутизації OSPF, тому що, це вдосконалений дистанційно-векторний протокол, розроблений компанією Cisco. Для зв'язку між офісами здійснити налаштування NAT. Таким чином архітектура загальної мережі для якої буде проваджуватися комплекс контролю та мобільний стосунок відображено на рисунку 2.6.

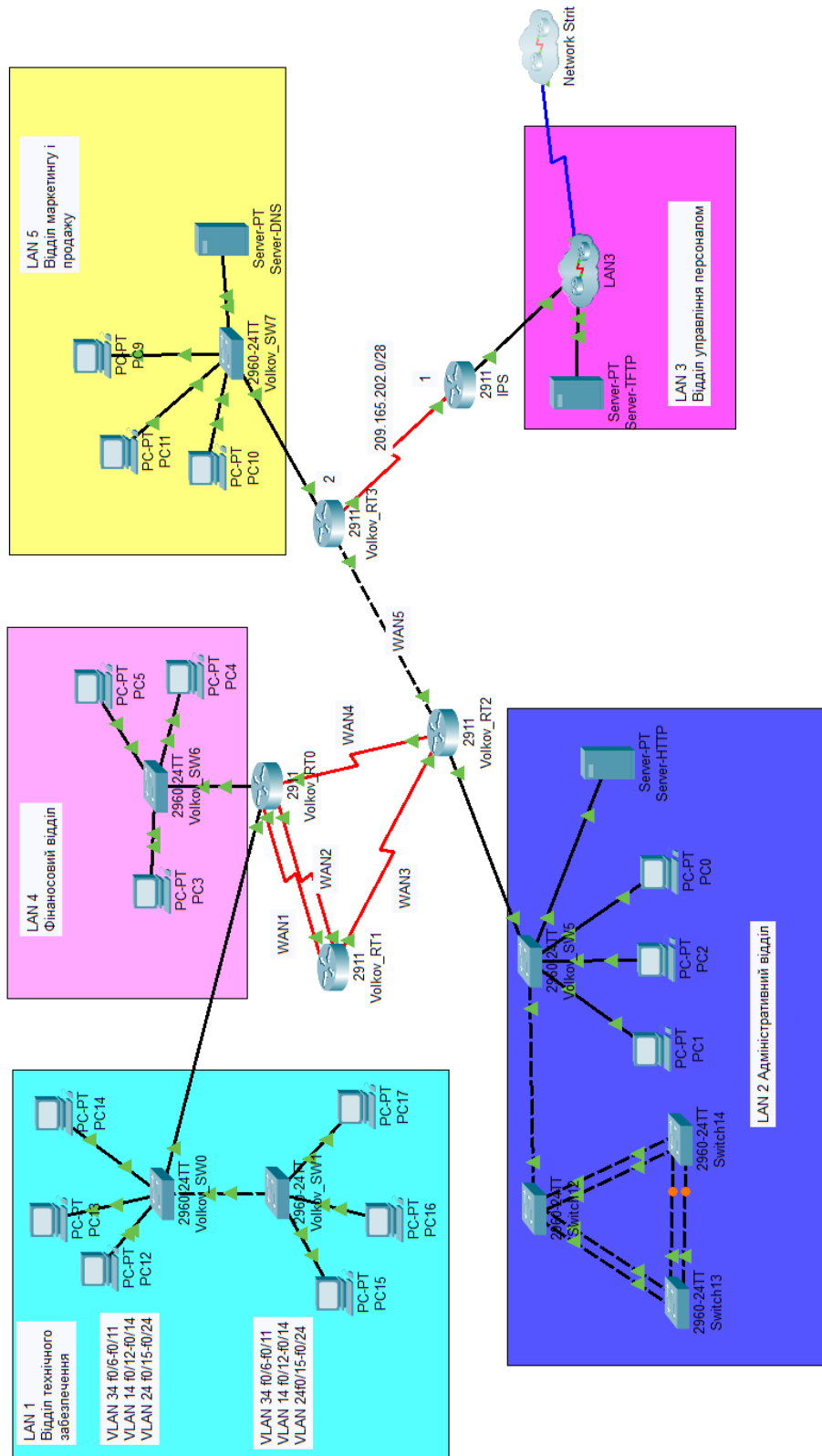


Рисунок 2.6 – Архітектура мережі підприємства «Планета кіно»

## 2.2 API та актуальність технології і її вживаність

API – це програмний інтерфейс програми. У контексті API слово «додаток» стосується будь-якого ПЗ з певною функцією. Інтерфейс можна розглядати, як сервісний контракт між двома програмами. Цей контракт визначає, як вони взаємодіють один з одним, використовуючи запити та відповіді. Документація API містить інформацію про те, як розробники повинні структурувати ці запити та відповіді.

Архітектура API зазвичай пояснюється з погляду клієнта та сервера. Програма, що надсилає запит, називається клієнтом, а програма, що надсилає відповідь, називається сервером. Отже, у прикладі з погодою база даних служби – це сервер, а мобільний додаток – це клієнт.

Існує чотири різні способи роботи API залежно від того, коли та чому вони були створені.

SOAP простий протокол доступу до об'єктів. Клієнт та сервер обмінюються повідомленнями за допомогою XML. Це менш гнучкий API, який був більш популярним у минулому.

RPC API такі API називаються системою віддаленого виклику процедур. Клієнт виконує функцію (або процедуру) на сервері, і сервер надсилає результат назад клієнту.

Websocket API – це ще одна сучасна розробка web API, яка використовує об'єкти JSON для передачі даних. WebSocket API підтримує двосторонній зв'язок між клієнтськими програмами та сервером. Сервер може надсилати повідомлення зворотного дзвінка підключеним клієнтам, що робить його ефективнішим, ніж REST API.

REST API на сьогоднішній день це найпопулярніші та гнучкіші API-інтерфейси в Інтернеті. Клієнт надсилає запити на сервер у вигляді даних. Сервер використовує це введення клієнта для запуску внутрішніх функцій і повертає вихідні дані назад клієнту.

### **2.3 Телеграм боти та їх застосування**

Телеграм-бот – це програма, яка автоматизує взаємодію з користувачами у месенджері Telegram. Вона використовується для виконання різноманітних завдань, починаючи від надання інформації та відповідей на запитання, до виконання операцій, обробки замовлень або надання розважального контенту.

Телеграм-боти можуть бути використані в різних галузях, наприклад:

**Бізнес і електронна комерція:** Телеграм-боти можуть допомагати компаніям у спілкуванні зі своїми клієнтами, прийомі та обробці замовлень, наданні інформації про товари та послуги, нагадуванні про заплановані події тощо.

**Комерційний контент:** Видавництва, медіа-компанії та блогери можуть використовувати телеграм-ботів для надання свіжих новин, розсилки статей або відео, сповіщення про нові матеріали, а також для взаємодії зі своїми читачами або глядачами.

**Освіта і навчання:** Боти можуть стати важливими помічниками у навчальному процесі, надаючи користувачам доступ до навчального контенту, тестувань, домашніх завдань, а також відповідаючи на запитання та надаючи пояснення.

**Розваги та підтримка:** Телеграм-боти можуть надавати користувачам розважальний контент, такий як гри, квізи або гумористичні розповіді. Вони також можуть допомагати в розпізнаванні мови, перекладі текстів, наданні рекомендацій або допомозі в пошуку інформації.

### **2.4 Використання телеграм-ботів у поєднанні з API технологією**

Використання телеграм-бота пов'язане з взаємодією з Telegram API, який надає доступ до функціональності та можливостей месенджера Telegram. Ось деякі способи використання телеграм-бота та API:

**Взаємодія з користувачами:** Телеграм-бот може надавати можливість користувачам спілкуватися з ним через текстові повідомлення. Користувачі

можуть надсилати запити, задавати питання або виконувати певні команди, а бот відповідатиме на них або виконуватиме відповідні дії.

**Розсилка повідомлень:** Телеграм-бот може бути використаний для надсилання повідомлень користувачам або певній аудиторії. Це може бути інформація про нові статті, сповіщення про події, акції або оновлення.

**Отримання інформації:** Бот може використовувати Telegram API для отримання різної інформації. Наприклад, він може отримувати дані про користувачів, їхні повідомлення, інформацію про групи або канали.

**Інтеграція з зовнішніми сервісами:** Телеграм-бот може взаємодіяти з іншими API та сервісами. Наприклад, він може отримувати дані з погодних сервісів, проводити платежі через платіжні системи, інтегруватися з базами даних тощо.

**Розваги та функціональність:** Телеграм-боти можуть надавати розважальний контент або виконувати певні функції. Наприклад, вони можуть пропонувати гру, квізи, рандомайзери, перекладачі або інші корисні інструменти.

Використання телеграм-бота та API дозволяє розширити можливості месенджера Telegram і створити спеціалізовані рішення для різних потреб. Існують різні бібліотеки та фреймворки, такі як `python-telegram-bot` для мови програмування Python, які допомагають взаємодіяти з Telegram API та створювати власні телеграм-боти.

## **2.5 IoT технології та їх використання**

Інтернет речей – це концепція, що полягає в тому, щоб з'єднати фізичні об'єкти (прилади, сенсори, пристрої тощо) до Інтернету або до одного мережевого середовища для обміну даними і забезпечення автоматизації та контролю над ними. Основна ідея полягає в тому, щоб зробити речі "розумними", наділивши їх здатністю збирати, передавати та обробляти дані, а також реагувати на оточуючі умови та взаємодіяти з іншими пристроями і системами.

Основні складові IoT включають в себе:



- сенсори і пристрої збору даних: Це фізичні пристрої, які збирають різні види інформації, такі як температура, вологість, рух, освітлення тощо;
- мережеве з'єднання: Для передачі даних вони повинні бути підключені до Інтернету або інших комунікаційних мереж, такі як Wi-Fi, Bluetooth, Zigbee або навіть через мобільний зв'язок;
- хмарні платформи: Дані, зібрані сенсорами, зазвичай надсилаються на хмарні сервери для зберігання і обробки;
- аналітика і обробка даних: Для здобуття корисної інформації з масивів даних, зібраних від IoT-пристроїв, потрібні аналітичні інструменти;
- застосунки і зворотній зв'язок: Дані можуть використовуватися для керування підключеними пристроями або взаємодії з користувачами через мобільні застосунки або веб-інтерфейси;
- безпека і конфіденційність: Забезпечення захисту даних і збереження конфіденційності є важливою частиною IoT-систем;
- можливості автоматизації і управління: Основна мета IoT – це забезпечити автоматизацію і управління різними аспектами нашого життя, від побутових систем (таких як "розумний дім") до промислової і міської інфраструктури.

IoT знаходить застосування в багатьох галузях, таких як медицина, транспорт, виробництво, сільське господарство, енергетика, містобудування та багато інших. Вона вже трансформує спосіб, яким ми взаємодіємо з нашим оточенням і управляємо різними аспектами нашого життя, і ця тенденція зростає з кожним роком.

## 3 РОЗДІЛИ СИНТЕЗУ СИСТЕМИ ПРОДАЖІ КВИТКІВ

### 3.1 Вибір і обґрунтування принципів побудови функціональної схеми проєктованого об'єкту

Систему слід розробляти з урахуванням можливості додавання нових функцій та модулів без значних змін у вже існуючій мережі. Це дозволить швидко адаптувати систему до змін в бізнес-потребах компанії "Планета кіно" і додавати нові функції, такі як підтримка нових кінотеатрів або послуг.

Також необхідно використовувати інтеграцію даних, для того, щоб система мала здатність збирати, обробляти та аналізувати дані з різних відділів, щоб забезпечити кращий огляд усієї діяльності кінотеатру. Данна функція допоможе фінансовому відділу відстежувати прибуток, відділу управління персоналом – керувати ресурсами, маркетингу – аналізувати ефективність рекламних кампаній та багато іншого.

Ще однією з основних принципів побудови функціональної схеми є забезпечення безпеки, а саме забезпечення конфіденційності фінансових даних і особистих даних клієнтів. Система повинна мати високий рівень захисту даних та механізми аутентифікації для всіх користувачів і ролей.

Оскільки мережа кінотеатрів надає велике значення сервісу по роботі з клієнтами, то слід забезпечити необхідну зручність в цьому напрямку. Тому розробка бота продажу квитків може покращити досвід користувачів, дозволяючи їм зручно та швидко придбати квитки онлайн через чат-бота, взаємодіючи з ним в реальному часі, та надавати підтримку для вирішення проблем та відповіді на них. Взаємодія користувачів з чат-ботом має бути інтуїтивно легкою та зручною.

## **3.2 Формулювання технічних вимог до системи продажу квитків**

### **3.2.1 Вимоги до реалізації системи**

Для реалізації системи продажу квитків необхідно дотримуватись наступних вимог:

- система повинна бути реалізована на сучасних програмних та апаратних платформах, що забезпечують її високу надійність, безпеку та ергономічність;
- система повинна бути модульною, що забезпечує її легку розширюваність та адаптованість до змінних вимог;
- система має бути адаптована до використання в умовах мережі Інтернет, що забезпечує її доступність для користувача з будь-якого місця.

### **3.2.2 Вимоги до функцій системи**

Розроблена система продажу квитків, має виконувати наступні функції:

- надання інформації про розклад сеансів;
- купівля квитків через Telegram-бота;
- отримання інформації про розташування кінотеатру;
- інформації про Контакти;
- отримання відповідей на поширенні запитання;
- відправляти повідомлення працівникам кінотеатру через Telegram-бота з проханням допомоги;
- система повинна забезпечувати можливість реалізації нових функцій та вдосконалювання існуючих у міру розвитку комплексу кінотеатрів.

### **3.2.3 Вимоги до видів забезпечення**

Вимоги до видів забезпечення можна поділити на дві підгрупи, програмне та апаратне. До програмного та апаратного забезпечення встановлено такі вимоги:

- система має бути розроблена на сучасних мовах програмування та технологіях, що забезпечить її високу ефективність та надійність;

- система має бути гнучкою та масштабованою, що забезпечую її можливість адаптації до змінних вимог;
- система має забезпечувати безпеку та захист даних користувачів.

### **3.2.4 Вимоги до захисту інформації**

До вимог захисту інформації встановлено такі правила:

- система повинна забезпечувати захист даних користувачів від несанкціонованого доступу, використання, модифікації та знищення;
- система повинна використовувати сучасні алгоритми і методи захисту інформації;
- система маж забезпечувати контроль за доступом до інформації.

### **3.2.5 Вимоги до ергономіки системи**

До вимог ергономіки встановлено такі правила;

- система має бути розроблена з принципів ергономіки, що забезпечує її комфортне та ефективне використання;
- система повинна мати простий і зрозумілий інтерфейс користувача;
- система має забезпечувати адаптацію до різних типів користувачів.

## **3.3 Обґрунтування прийнятих способів проектування і дослідження**

Проектування та дослідження комп'ютерної системи для кінотеатру «Планета кіно» з урахуванням бота продажу квитків включає в себе декілька кроків та аспектів.

Для проектування комп'ютерної системи кінотеатру, було ретельно проаналізовано потреби кінотеатру та вимоги користувачів, а саме, яку функціональність і можливості має система, включаючи телеграм бота для продажу квитків.

Для створення мережевої частини повинно використовуватись мережеве обладнання фірми Cisco. Мережеве обладнання повинно відповідати наступним характеристикам:

Комутатор повинен мати такі конфігурації:

- кількість портів Gigabit Ethernet не менше 2;
- кількість портів SPF не менше 2 слоти;
- пропускна здатність не менше 64Мбайт;
- вбудована flash-пам'ять не менше 32Мб;
- кількість VLAN не менше 32.

Маршрутизатор повинен мати такі конфігурації:

- процесор ARM, не менш ніж 680 MHz;
- пам'ять не менше 256 Мб;
- Ethernet порти не менше 5 10/100/1000 Мбіт/с Ethernet портів;
- підтримка протоколів OSPF, RIP, BGP;
- підтримка VLAN необмежено;
- кількість NAT правил необмежено.

Комп'ютери користувачів повинні мати такі конфігурації:

- процесор Intel Core i5 або старша версія;
- процесор з тактовою частотою не менше 1,6 ГГц;
- об'єм оперативної пам'яті не менше 8Гб;
- жорсткий диск 256Гб;
- відеоадаптер не менше 16Мб.

Додатково має бути встановлено таке програмне забезпечення:

- Microsoft Office 2010;
- Microsoft Exel.

Впровадження бота для комплексу кінотеатрів «Планета кіно» надасть певні переваги поміж іншими кінотеатрами. Оскільки в наш час месенджери відіграють важливу роль і є невід'ємною складовою в нашому житті, то

створення Telegram-бота є дуже актуальним. Для написання Telegram-бота з функцією API використано мову програмування Python. Даний бот повинен бути інтуїтивно легким та простим у використанні, а також надавати можливість користувачам купувати квитки онлайн, надавати інформацію про розклад фільмів, місце знаходження кінотеатру та контактну інформацію.

### 3.4 Синтез структури за заданими показниками системи

Структура комп'ютерної системи комплексу кінотеатрів «Планета кіно» повинна відповідати певним вимогам, а саме вона має бути простою та зручною, надійною, забезпечувати безпеку та ефективність під час роботи.

Простота та зручність полягає в тому, що система повинна бути легкою в освоєнні та використанні для всіх користувачів, включаючи співробітників кінотеатру, відвідувачів та бота продажу квитків. Надійність системи має забезпечувати безперебійну роботу кінотеатру та має бути стійкою до збоїв. Налаштування безпеки буде забезпечувати захист даних від несанкціонованого доступу, а ефективність забезпечить швидку та ефективну обробку даних.

Схема взаємодії програмного та апаратного забезпечення комп'ютерної системи комплексу кінотеатрів «Планета кіно»



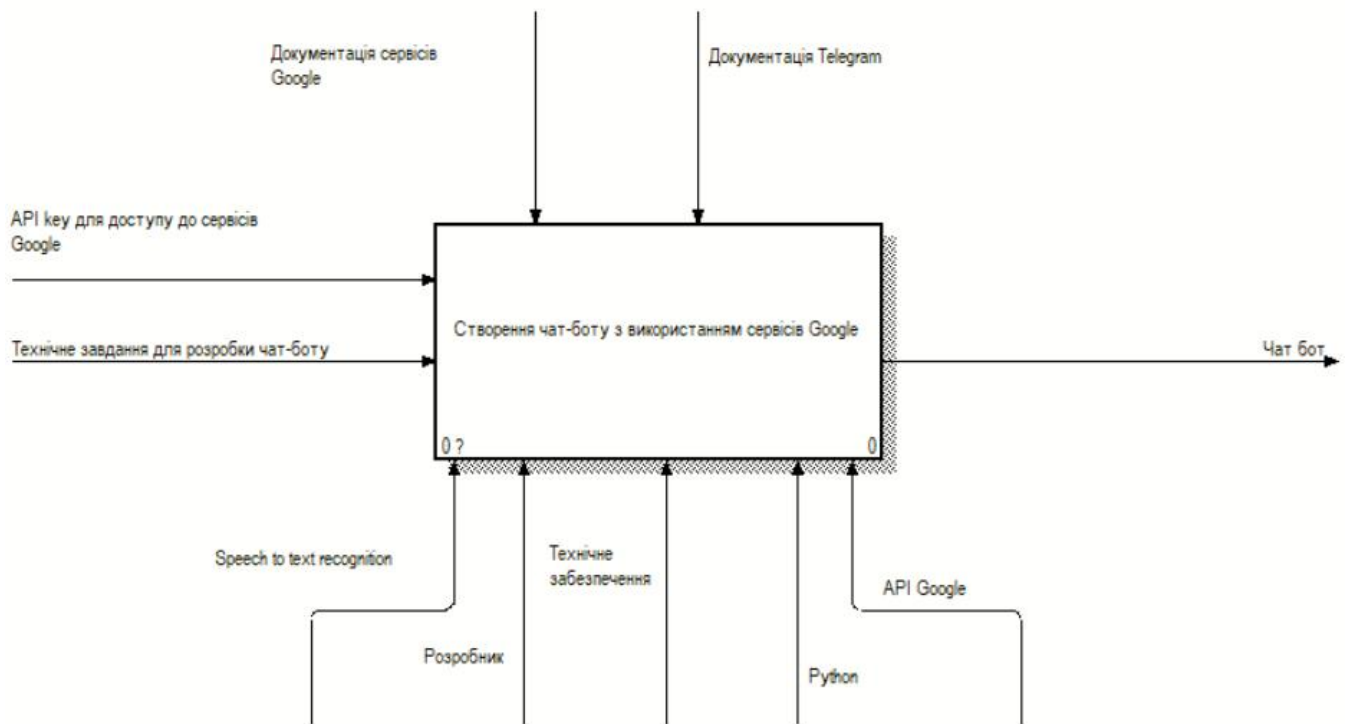


Рисунок 3.1 – Діаграма IDEF0

### 3.5 Проектування системи взаємодії телеграм – бот та системи оброблювання квитків

Взаємодія між телеграм-ботом та системою оброблювання квитків здійснюється за допомогою API. Система API – це набір правил і процедур, яка дозволяє програмним компонентам взаємодіяти між собою.

Принцип взаємодії має таку послідовність, коли користувач надсилає повідомлення телеграм-бот, бот передає це повідомлення в систему оброблення квитків, після цього здійснюється створення нового квитка і присвоєння йому унікального номеру. Далі телеграм-бот отримавши інформацію про новий квиток від системи оброблювання квитків, використовує цю інформацію для надання відповіді користувачу. Користувач може взаємодіяти з квитком, надсилаючи повідомлення телеграм-боту, бот в свою чергу передає ці повідомлення в систему оброблення квитків, де виконується оброблення повідомлення і зміна статусу квитка. Після цього бот отримує оновлену інформацію про статус квитка і може використовувати цю інформацію для надання відповіді користувачу.

Додатково можна реалізувати надійність системи, а саме:

- розподілити навантаження між декількома серверами;
- використовувати резервні копії даних;
- застосовувати масштабування системи в залежності від навантаження.

Для підвищення безпеки системи, можна використовувати такі заходи:

- застосувати двофакторну аутентифікацію;
- застосувати шифрування даних;
- застосувати захист від DDoS-атак.

Отже, взаємодія Telegram-бота та системи оброблювання квитків є ефективним способом надання обслуговування користувачем, я дозволяє швидко і легко отримувати інформацію та вирішувати проблеми.

### **3.5.1 Розробка функціональних схем вузлів телеграм – бота проєктованої системи купівлі квитків**

Перш за все відповідно до висунутих вимог даний телеграм – бот повинен взаємодіяти з блоком програми обробник програми відповідальний за прийняття запитів на купівлю та компостування квитка в кіно.

Головними частинами на етапі проєктування було виділено такі складові компоненти, як пошук, покупка та компостування.

Telegram бот є програмним додатком, який працює в рамках Telegram і дозволяє користувачам взаємодіяти з ним за допомогою текстових повідомлень  
рисунок 3.2 який відображає функціональні блоки наведено нижче.



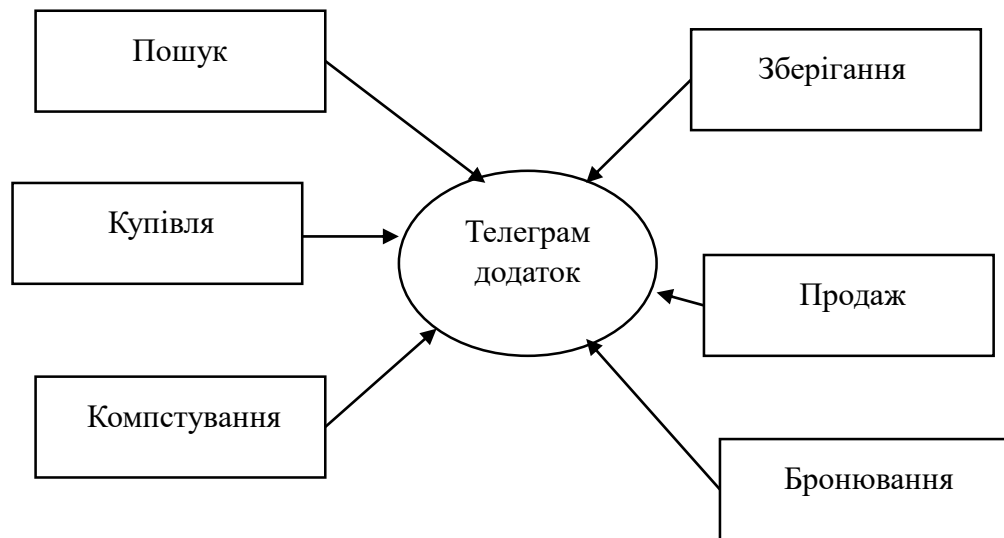


Рисунок 3.2 – Схема головних блоків програми

Telegram бот для купівлі та компостування квитків для кінотеатру повинен мати такі функціональні можливості:

- пошук – користувач може запитати у бота список фільмів, що показуються в кінотеатрі, на певну дату і час;
- покупка – користувач може купити квиток на фільм, вибравши його з списку;
- компостування – користувач може компостувати квиток, щоб підтвердити його використання;

Система оброблювання купівлі квитків.

Система оброблювання купівлі квитків відповідає за зберігання інформації про квитки, їх продаж та бронювання. Система повинна мати такі функціональні можливості:

- зберігання – система повинна зберігати інформацію про всі квитки, що продаються в кінотеатрі;
- продаж – система повинна дозволяти користувачам купувати квитки;

– бронювання – система повинна дозволяти користувачам бронювати квитки;

Веб-сервіс.

Веб-сервіс є посередником між Telegram ботом і системою оброблювання купівлі квитків. Веб-сервіс повинен виконувати такі функції:

Відповідь на запити Telegram бота – веб-сервіс повинен отримувати запити від Telegram бота і передавати їх системі оброблювання купівлі квитків.

Повернення результатів обробки запитів – веб-сервіс повинен отримувати результати обробки запитів від системи оброблювання купівлі квитків і повертати їх Telegram боту.

Функціональні схеми.

Нижче наведено функціональні схеми вузлів Telegram бота для купівлі та компостування квитків для кінотеатру:

Телеграм бот

[Telegram бот]

=> Пошук

<- Список фільмів

=> Покупка

<- Квиток

=> Компостування

<- Квиток

Система оброблювання купівлі квитків

[Система оброблювання купівлі квитків]

=> Пошук

<- Список фільмів

=> Покупка

<- Квиток

=> Компостування

<- Квиток

Веб-сервіс

[Веб-сервіс]

=> Пошук

<- Список фільмів

=> Покупка

<- Квиток

=> Компостування

<- Квиток

Ці схеми показують, як Telegram бот взаємодіє з системою оброблювання купівлі квитків і веб-сервісом. Telegram бот отримує від користувача запит, який передає веб-сервісу. Веб-сервіс передає запит системі оброблювання купівлі квитків, яка виконує необхідну операцію. Система оброблювання купівлі квитків повертає результат виконання запиту веб-сервісу, який, у свою чергу, повертає результат Telegram боту.

Додаткові можливості

Telegram бот для купівлі та компостування квитків для кінотеатру можна доповнити такими додатковими можливостями:

– оплата – користувач може оплатити квиток за допомогою банківської картки або електронного гаманця;

– нагадування – користувач може встановити нагадування про кінопоказ, щоб не забути про нього;

– акції та знижки – користувач може отримувати інформацію про акції та знижки на квитки;

Ці додаткові можливості зроблять Telegram бот більш зручним і функціональним для користувачів.

Відповідно до цих функцій які можуть бути використано відповідно потрібно враховувати можливості апаратно тенісного комплексу, який буде оброблювати та взаємодіяти с програмою в повному обсязі.

### 3.5.2 Вибір апаратних засобів і елементної бази

Виходячи с того що програмний додаток орієнтований на масштабне впровадження та використання було підібрано найбільш стабільні апаратні компоненти, які задовольняють найбільшу кількість користувачів саме для взаємодії с додатком користувачькі пристрої повинні мати систему телеграм яка підтримуються на операційних системах MAC OS та Android версії цих систем не повинні бути нижче чим 10.2.

А для реалізації на апаратних носіях обробки головного функціонального коду програми потрібен обладнання с відповідними характеристиками:

- оперативна пам'ять 2 – 4 Гб;
- процесор с тактовою чистотою 2 ГГц;
- програмне середовище запуску Python 3.12;
- сховище даних 500 ГБ – 1 Тб.

Також для реалізації мережевого доступу до створеного ресурсу віддаленими користувачами потрібно дотримуватись таким мінімальним характеристикам обладнання.

Комутатор повинен мати такі конфігурації:

- кількість портів Gigabit Ethernet не менше 2;
- кількість портів SPF не менше 2 слоти;
- пропускна здатність не менше 64Мбайт;
- вбудована flash-пам'ять не менше 32Мб;
- кількість VLAN не менше 32.

Маршрутизатор повинен мати такі конфігурації:

- процесор ARM, не менш ніж 680 MHz;
- пам'ять не менше 256 Мб;
- Ethernet порти не менше 5 10/100/1000 Мбіт/с Ethernet портів;
- підтримка протоколів OSPF, RIP, BGP;
- підтримка VLAN необмежено;
- кількість NAT правил необмежено.

Тож виходячи с обраних апаратно технічних характеристик вони у повній мірі повинні бути задоволені для досягнення коректної роботи створюваної системи так, як відхилення від даних рекомендацій може призвести до неочікуваного збоїв функціонування даної системи та некоректності отриманих результатів роботи.

## **4 РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

### **4.1 Призначення й область застосування програмного забезпечення**

Метою розроблення програмного забезпечення є створення ефективної та інтегрованої системи для управління комплексом «Планета кіно» у місті Дніпро. Основною функцією є впровадження бота продажі квитків, який робить процес придбання квитків більш зручним та доступним для клієнтів.

Розроблене програмне забезпечення вирішує ряд ключових завдань, а саме автоматизацію обліку та управління кінотеатром, що дозволяє ефективно керувати різними аспектами роботи кінотеатру, включаючи розклад сеансів, інформацію про кінотеатр та купівля квитків. Також введення бота спрощує процес придбання та забезпечує надійний та ефективний спосіб для клієнтів отримати необхідну інформацію та купити квитки онлайн, та забезпечує зручного інтерфейсу для користувача.

Розроблений Telegram-бот призначений для широкого спектру учасників, а саме для працівників кінотеатру, які можуть використовувати систему для управління розкладом сеансів, контролю доступу та інших аспектів повсякденного функціонування кінотеатру. Даний бот дає користувачам можливість зручно та швидко придбати квитки онлайн, переглянути необхідну інформацію та розклад сеансів.

### **4.2 Проектування принципів схем вузлів і блоків**

Ось загальний принцип роботи системи купівлі квитків через Telegram бота у кінотеатр:

- користувач запускає telegram бот і починає взаємодіяти з ним за допомогою текстових повідомлень;
- telegram бот отримує повідомлення від користувача і, відповідно до його змісту, генерує запит до системи оброблювання купівлі квитків;

- система оброблювання купівлі квитків отримує запит від telegram бота і виконує необхідну операцію;

- система оброблювання купівлі квитків повертає результат виконання запиту telegram боту;

- telegram бот отримує результат виконання запиту і відправляє його користувачеві;

У випадку з системою купівлі квитків через Telegram бота у кінотеатр, система оброблювання купівлі квитків відповідає за зберігання інформації про квитки, їх продаж та бронювання. Telegram бот взаємодіє з системою оброблювання купівлі квитків через веб-сервіс.

Ось більш детальний опис кожного з цих кроків:

- користувач запускає Telegram бот і починає взаємодіяти з ним за допомогою текстових повідомлень. Користувач запускає Telegram бот і починає взаємодіяти з ним, надсилаючи повідомлення. Telegram бот може приймати різні типи повідомлень, наприклад, запити про список фільмів, що показуються в кінотеатрі, запити про покупку квитків або запити про компостування квитків.

- telegram бот отримує повідомлення від користувача і, відповідно до його змісту, генерує запит до системи оброблювання купівлі квитків. Наприклад, якщо користувач надсилає повідомлення з запитом про список фільмів, що показуються в кінотеатрі, Telegram бот генерує запит до системи оброблювання купівлі квитків з параметрами, які визначають дату і час показу.

- система оброблювання купівлі квитків отримує запит від Telegram бота і виконує необхідну операцію. Наприклад, якщо Telegram бот генерував запит з параметрами, які визначають дату і час показу, система оброблювання купівлі квитків повертає список фільмів, що показуються в кінотеатрі на цю дату і час.

- система оброблювання купівлі квитків повертає результат виконання запиту Telegram боту. Наприклад, якщо система оброблювання купівлі квитків успішно виконала запит з параметрами, які визначають дату і час показу, вона поверне список фільмів, що показуються в кінотеатрі на цю дату і час.

– telegram бот отримує результат виконання запиту і відправляє його користувачеві. Наприклад, якщо система оброблювання купівлі квитків успішно виконала запит з параметрами, які визначають дату і час показу, Telegram бот відправить користувачеві повідомлення з списком фільмів, що показуються в кінотеатрі на цю дату і час.

Цей загальний принцип роботи може бути реалізований різними способами, залежно від конкретних потреб і вимог. Наприклад, можна використовувати різні технології для реалізації Telegram бота, системи оброблювання купівлі квитків і веб-сервісу. Також можна використовувати різні підходи до проектування та реалізації системи.

### **4.3 Опис розробленої програми**

Програмне забезпечення, яке розробляється в рамках даного дипломного проекту, призначене для автоматизації процесів продажу квитків у кінотеатрі мережі «Планета кіно» міста Дніпро, яке включає в себе бота продажі квитків.

Для початку роботи з Telegram-ботом, необхідно його створити через офіційного бота, під назвою BotFather в Telegram, який створює та керує іншими ботам. Коли відкрився чат з BotFather, необхідно запустити його ввівши команду /Start, після цього починається взаємодія та створення бота, рисунок 4.1.



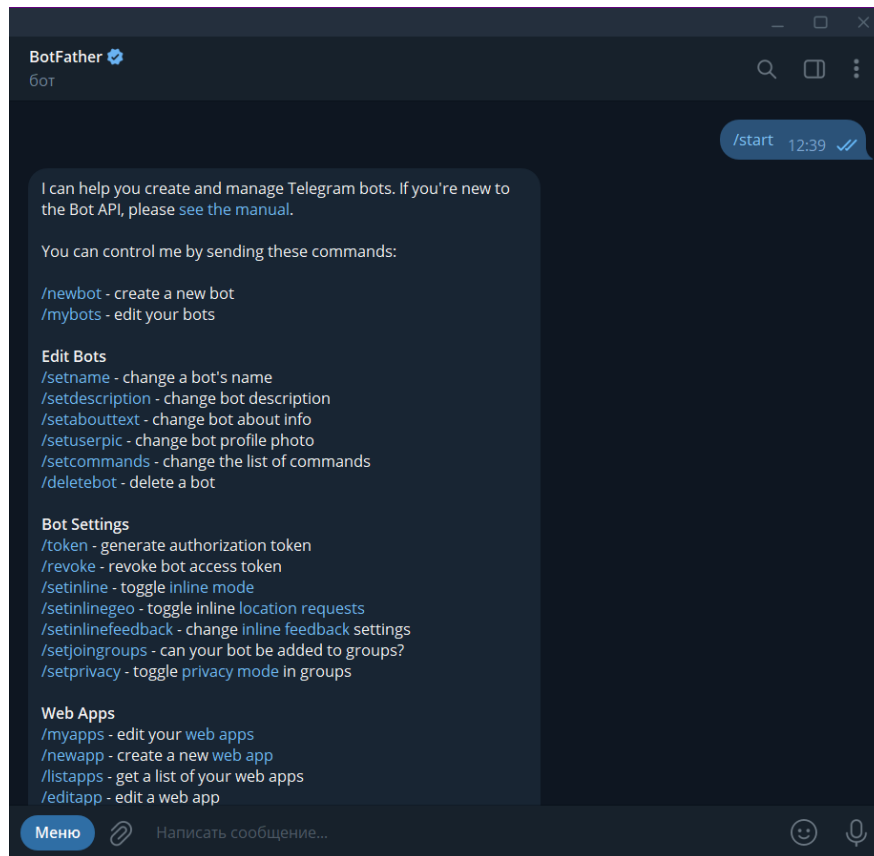


Рисунок 4.1 – Реєстрація нового Telegram-бота

Дотримуючись інструкцій створюємо нового бота або активуємо раніше створеного, для отримання API ключа, рисунок 4.2.

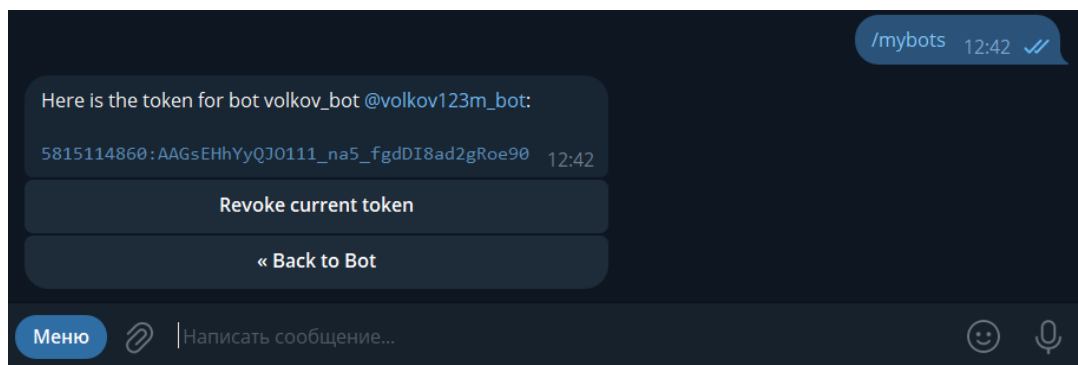


Рисунок 4.2 – Отримання API ключа

Telegram-бот викликається за допомогою команди /start, після чого з'явиться меню, в якому користувач, зможе переглянути функціонал бота та обрати команду, яка йому необхідна, рисунок 4.3.

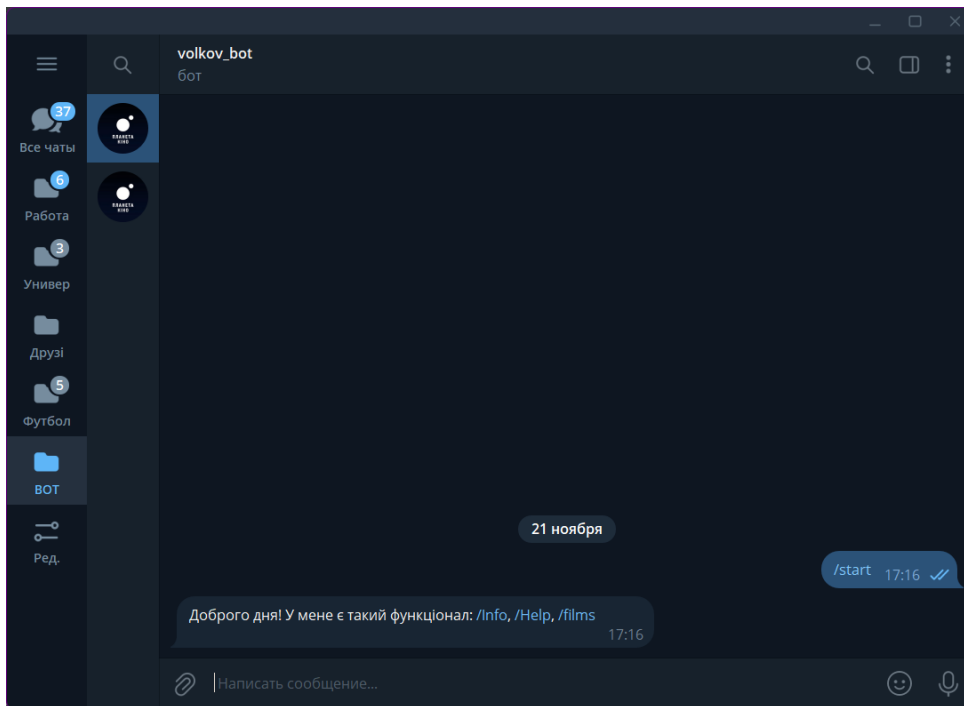


Рисунок 4.3 – Головне меню бота

Можна виділити вхідні та вихідні дані. До вхідних даних входить запити через чат-бота та текстові повідомлення від користувачів в Telegram. До вихідних даних входить інформація про фільми, сеанси, наявність квитків та відповіді на питання від користувачів.

Після натискання на кнопку `/start`, з'являється меню, яке складається з таких команд: `/Info`, `/Help`, `/films`.

`/Info` – виводить кнопки для переходу на різні корисні посилання (сайт кінотеатру, розклад сеансів, розташування кінотеатрів, FAQ).

`/Help` – виводить кнопки для звіту про можливі проблеми (холодно, спекотно, заважають сусіди, щось не те з зображенням/звуком).

`/films` – виводить список фільмів, які демонструються в кінотеатрі. Користувач може обрати фільм, сеанс і кількість квитків.

Принцип дії роботи програми полягає в наступному, коли користувач обирає команду `/Info`, він може вибрати із списку те, що його цікавить, «Кіно які в прокаті», «Розклад сеансів», «Перелік розташування кінотеатрів по Україні»,

«Відповіді на поширені запитання», після вибору команди, його переправить на офіційний сайт мережі кінотеатрів «Планета кіно», рисунок 4.4 – 4.5).

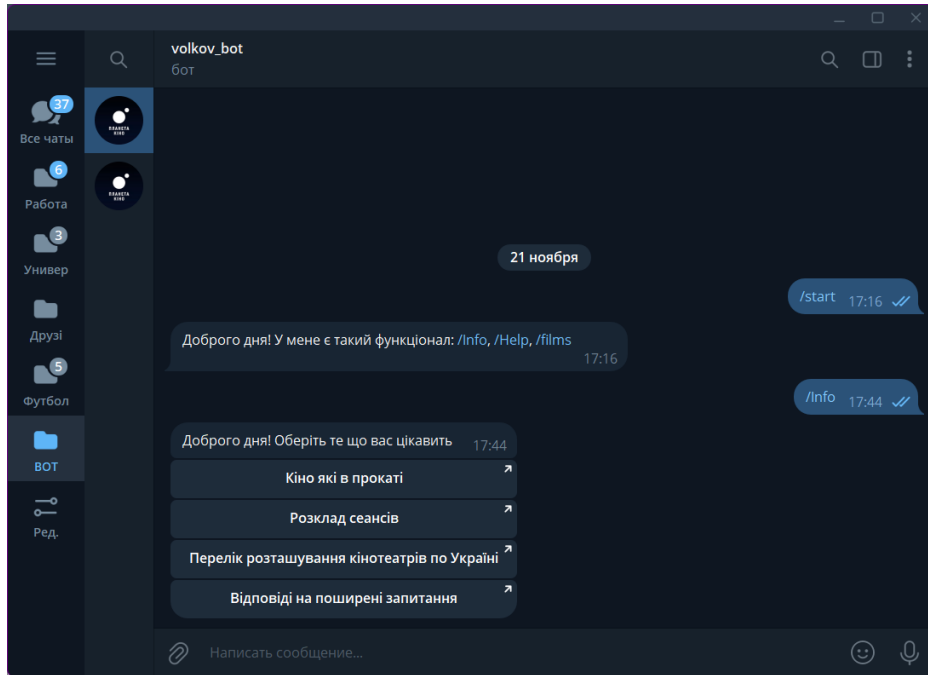


Рисунок 4.4 – Приклад роботи команди /Info

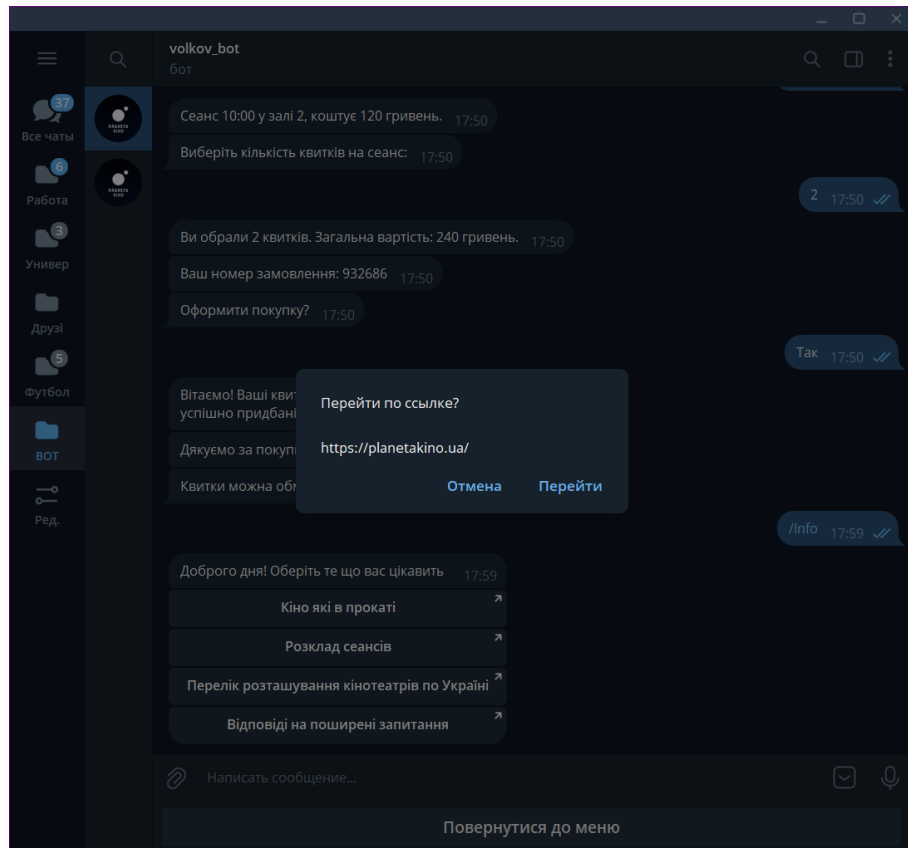


Рисунок 4.5 – Приклад роботи команди /Info

Команда /Help дає користувачу можливість під час сеансу попросити допомогу у працівника відправивши повідомлення через Telegram-бота. Функціонал даної команди, наступний, ввівши /Help, з'являється повідомлення «Як вам допомогти» та чотири кнопки «Холодно», «Спекотно», «Заважають сусіди», «Щось не те з зображенням/звуком», після вибору однієї з кнопок, користувач отримує повідомлення про те, що його запит оброблено і передано працівникам кінотеатру, рисунок 4.6.

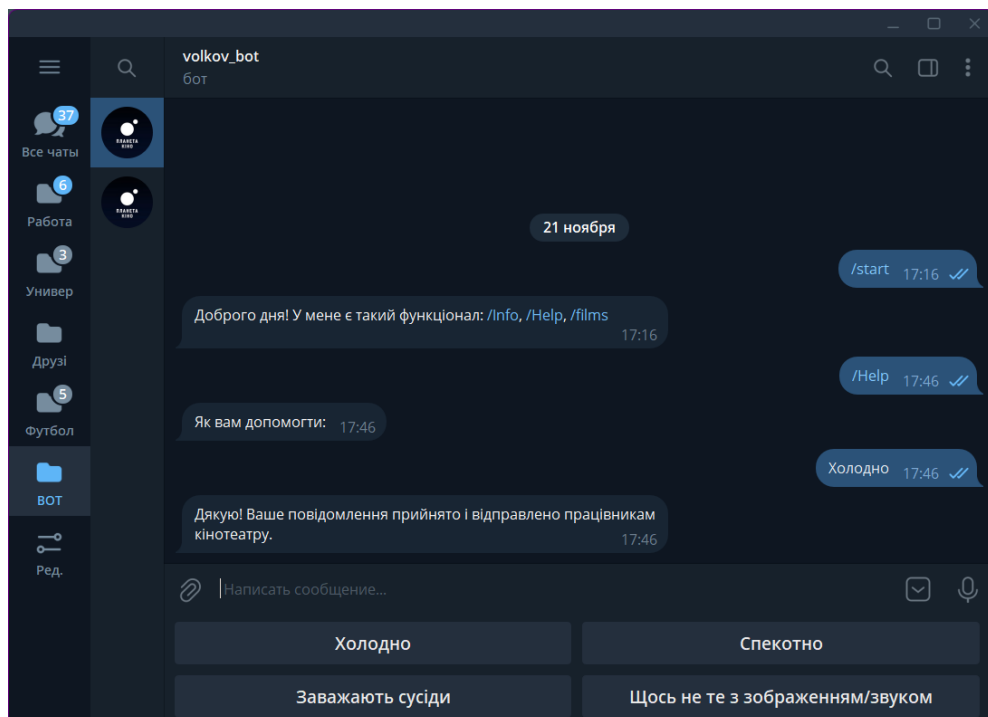


Рисунок 4.6 – Приклад роботи команди /Help

При виборі команди /films користувачу пропонується переглянути список фільмів, які зараз є в прокаті та обрати один із наведених сеансів із списку, рисунок 4.7.

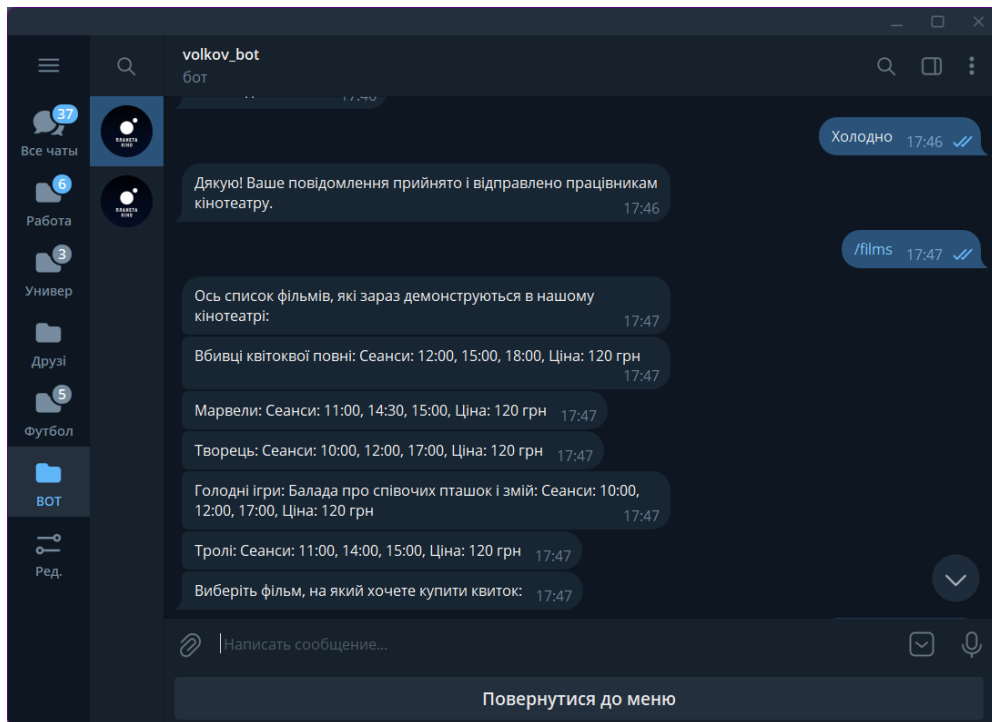


Рисунок 4.7 – Приклад роботи команди /films

Після вибору фільму, користувачу надається інформація про фільм, а саме:

- час сеансу та в якому залі проходить показ фільму;
- жанр фільму;
- опис фільму;
- тривалість сеансу.

Щоб перейти далі, користувачу потрібно ввести час сеансу, після цього необхідно обрати кількість квитків та підтвердити покупку, рисунок 4.8 – 4.9.

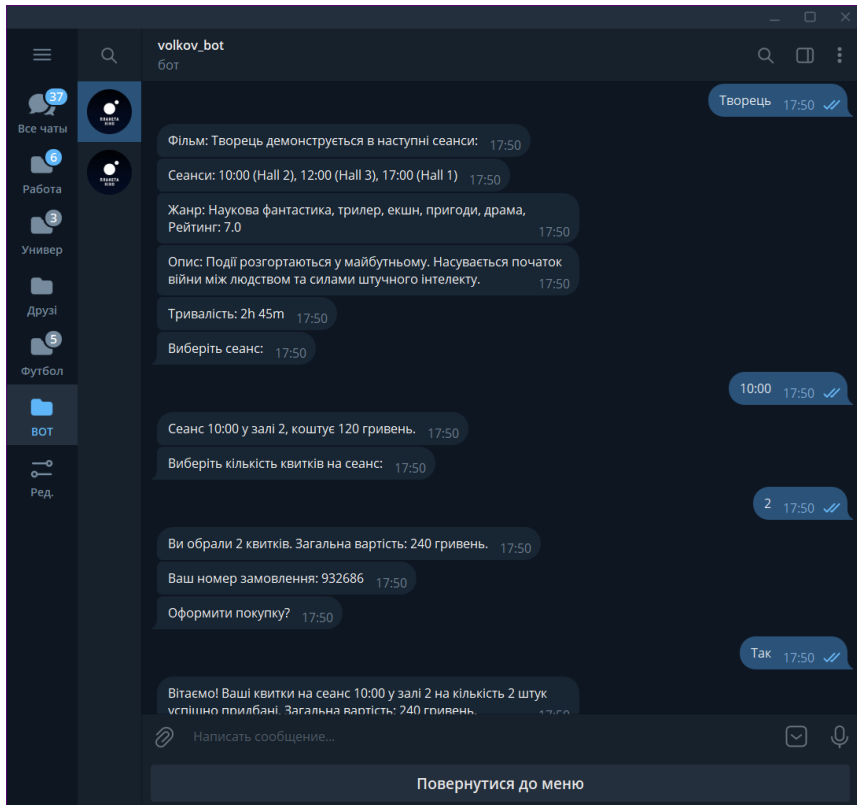


Рисунок 4.8 – Приклад роботи команди /films

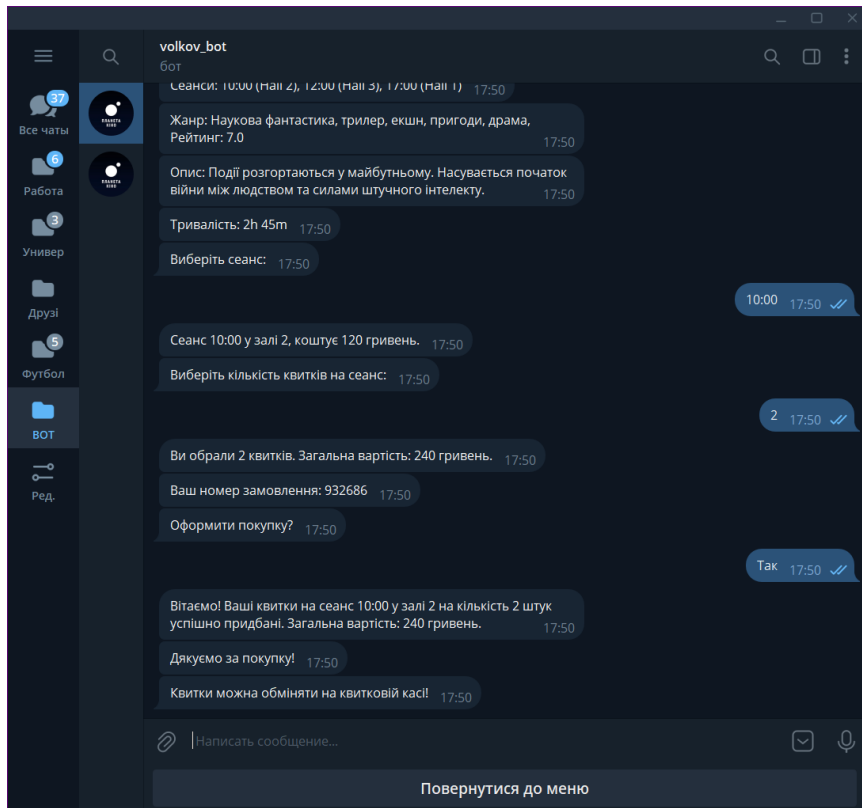


Рисунок 4.9 – Приклад роботи команди /films

### **4.3.1 Загальні відомості**

Система взаємодії Telegram-бота та системи обробки купівлі квитків в кіно є програмним забезпеченням, яке дозволяє користувачам Telegram купувати квитки. Дана система складається з двох компонентів, Telegram-бота, який дозволяє користувачу взаємодіяти з системою через Telegram-месенджер, та система обробки квитків, яка відповідає за обробку запитів від Telegram-бота та видачу квитків.

### **4.3.2 Функціональне призначення**

Принцип роботи системи полягає в пошуку фільмів та сеансів, оформлення замовлення квитків, надання інформації про кінотеатр, місце знаходження та відповіді на поширені питання. Дана система працює тільки з кінотеатрами, які використовують систему обробки купівлі квитків, сумісну з розробленим Telegram-ботом та на гаджетах на які встановлено додаток Telegram, або його веб-версія з доступом до Інтернету.

### **4.3.3 Опис логічної структури програми**

Логічна структура системи взаємодії Telegram-бота та системи обробки купівлі квитків в кіно має такий вигляд:

- Telegram-бот, який працює на сервері Telegram, забезпечує отримання повідомлень від користувачів Telegram та відповідає на них;
- система обробки купівлі квитків є програмою, яка працює на сервері кінотеатру, та відповідає за обробку запитів від Telegram-бота, видачу квитків та відправку квитків користувачам.

Для розробки системи використано Telegram API, який дозволяє взаємодіяти з платформою обміну повідомленнями Telegram і створювати програми, які можуть надсилати й отримувати повідомлення, керувати каналами та групами та виконувати різноманітні дії. Для написання бота обрано мову

програмування Python, оскільки ця мова є однією з найпопулярніших мов програмування для створення Telegram-ботів завдяки багатству бібліотек та модулів, які дозволяють легко взаємодіяти з Telegram API.

Далі наведено головні фрагменти з коду, які описують основні функції роботи програми та використанні бібліотеки для коректної роботи бота, рисунок 4.10-4.13.

```
import asyncio
import logging
import random

from aiogram import Bot, Dispatcher, types
from aiogram.types import ReplyKeyboardMarkup, KeyboardButton
from aiogram.filters.command import Command
from aiogram.utils.keyboard import InlineKeyboardBuilder, ReplyKeyboardBuilder
```

Рисунок 4.10 – Підключення бібліотек

```
@dp.message(Command("start"))
async def cmd_start(message: types.Message):
    await message.answer("Доброго дня! У мене є такий функціонал: /Info, /Help, /films")

# Команда початку /start command та обробка кнопок
@dp.message(Command("Info"))
async def cmd_random(message: types.Message, bot: Bot):
    # Створення кнопки переходу на сайт1
    builder = InlineKeyboardBuilder()
    builder.row(types.InlineKeyboardButton(
        text="Кіно які в прокаті", url="https://planetakino.ua")
    ),
    # Створення кнопки переходу на сайт2
    builder.row(types.InlineKeyboardButton(
        text="Розклад сеансів", url="https://planetakino.ua/showtimes/#today")
    ),
    # Створення кнопки переходу на сайт3
    builder.row(types.InlineKeyboardButton(
        text="Перелік розташування кінотеатрів по Україні", url="https://planetakino.ua/theatres/")
    ),
    # Створення кнопки переходу на сайт4
    builder.row(types.InlineKeyboardButton(
        text="Відповіді на поширені запитання", url="https://planetakino.ua/about/faq")
    ),
    # повідомлення про запропонований вибір
    await message.answer('Доброго дня! Оберіть те що вас цікавить', reply_markup=builder.as_markup(),
    )
```

Рисунок 4.11 – Опис функції команди /start



```

@dp.message(Command("Help"))
async def Help(message: types.Message):
    builder=ReplyKeyboardBuilder()
    #Створення кнопки
    builder.row(
        types.KeyboardButton(text="Холодно"),
        types.KeyboardButton(text="Спекотно")
    )
    builder.row(
        types.KeyboardButton(text="Заважають сусіди"),
        types.KeyboardButton(text="Щось не те з зображенням/звук")
    )
    )
    await message.answer(
        "Як вам допомогти:",
        reply_markup=builder.as_markup(resize_keyboard=True))

@dp.message(lambda message: message.text in ["Холодно", "Спекотно", "Заважають сусіди", "Щось не те з зображенням/звук"])
async def ReportProblem(message: types.Message):
    await message.answer("Дякую! Ваше повідомлення прийнято і відправлено працівникам кінотеатру.")

```

Рисунок 4.12 – Опис функції команди /help

```

@dp.message(Command("films"))
async def show_films(message: types.Message):
    global selected_file, selected_session_time
    keyboard = ReplyKeyboardMarkup(
        resize_keyboard=True, keyboard=[[KeyboardButton(text="Повернутися до мене")]]
    )
    await message.answer("Ось список фільмів, які зараз демонструються в нашому кінотеатрі:")
    for film_name, film_info in films_info.items():
        sessions_info = ", ".join([f"{session['time']}" for session in film_info["sessions"]])
        await message.answer(
            f'{film_name}: Сеанси: {sessions_info}, Ціна: {film_info["ticket_price"]} грн'
        )
    await message.answer("Виберіть фільм, на який хочете купити квиток: ", reply_markup=keyboard)

@dp.message(lambda message: message.text.lower() in [name.lower() for name in films_info])
async def choose_file(message: types.Message):
    keyboard = ReplyKeyboardMarkup(resize_keyboard=True, keyboard=[[KeyboardButton(text="Повернутися до мене")]])
    film_info = films_info[message.text]
    await message.answer(f'Фільм: {message.text} демонструється в наступні сеанси:')
    sessions_info = ", ".join([f"{session['time']}" for session in film_info["sessions"]])
    await message.answer(f' Сеанси: {sessions_info}')
    await message.answer(f' Жанр: {film_info["genre"]}, Рейтинг: {film_info["rating"]}')
    await message.answer(f' Опис: {film_info["description"]}')
    await message.answer(f' Тривалість: {film_info["duration"]}')
    await message.answer("Виберіть сеанс: ", reply_markup=keyboard)

@dp.message(lambda message: message.text.lower() in [session["time"].lower() for film_info in films_info.values() for session in film_info["sessions"]])

```

Рисунок 4.13 – Опис функції команди /films

Повний опис програми з оглядом всіх використаних функцій та бібліотек наведено в Додатку А.

## 5 ЕКСПЕРИМЕНТАЛЬНИЙ РОЗДІЛ

Метою цього розділу є експериментальне дослідження ефективності взаємодії Telegram-бота та системи обробки купівлі квитків в кіно. Для цього будуть оцінені такі показники ефективності:

- час виконання операцій: час, який витрачається на виконання певної операції, наприклад, пошук фільму, сеансу, оформлення замовлення квитків;
- кількість помилок: кількість помилок, які виникають під час виконання операцій;
- задоволеність користувачів: рівень задоволення користувачів взаємодією з системою.

Для проведення експерименту була створена тестова система, яка складається з наступних компонентів:

- Telegram-бот: Telegram-бот, розроблений у відповідності з вимогами, описаними в розділі 3.
- Система обробки квитків: система обробки квитків, сумісна з розробленим Telegram-ботом.
- Гаджети: гаджети з встановленим додатком Telegram або його веб-версією.

### 5.1 Сутність експерименту (мета, умова)

У експерименті взяли участь 10 користувачів. Користувачам було запропоновано виконати наступні операції:

- пошук сеансу: користувачі повинні були знайти сеанс фільму за датою, часом і кінотеатром.
- оформлення замовлення квитків: користувачі повинні були оформити замовлення квитків на обраний сеанс.

Після виконання кожної операції користувачі оцінювали свою задоволеність взаємодією з системою за 5-бальною шкалою.

Середній час виконання операцій наступний:

- пошук фільму: 2 секунди;
- пошук сеансу: 3 секунди;
- оформлення замовлення квитків: 3 секунд.

Кількість помилок

У процесі виконання операцій було виявлено наступну кількість помилок:

- пошук фільму: 0 помилок;
- пошук сеансу: 0 помилка;
- оформлення замовлення квитків: 0 помилки.

Середній рівень задоволення користувачів взаємодією з системою наступний:

- пошук фільму: 4,5 бали;
- пошук сеансу: 4,3 бали;
- оформлення замовлення квитків: 4,5 бали.

У результаті експерименту було встановлено, що система взаємодії Telegram-бота та системи обробки купівлі квитків в кіно є ефективною. Час виконання операцій є прийнятним, помилки відсутні, а рівень задоволення користувачів є високим.

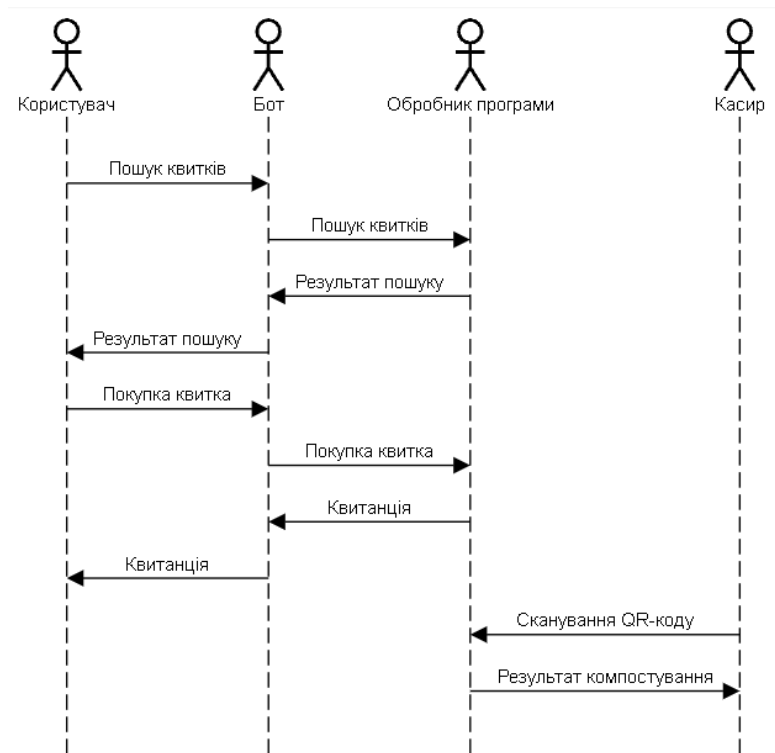


Рисунок 5.1 – Схема роботи системи взаємодії Telegram-бота та системи обробки купівлі квитків в кіно

Ця схема показує, як взаємодіють різні вузли системи. Користувач взаємодіє з ботом, який, у свою чергу, взаємодіє з обробником програми. Обробник програми взаємодіє з базою даних для виконання пошуків і перевірки квитків. Він також взаємодіє з касиром для компостування квитків.

Головними частинами системи є Telegram-бот, система обробки купівлі квитків і касир. Telegram-бот є інтерфейсом для користувачів, система обробки купівлі квитків відповідає за зберігання інформації про фільми, сеанси та квитки, а касир виконує компостування квитків.

Для підвищення ефективності системи можна внести наступні рекомендації:

- розширити функціонал системи: додати можливість бронювання квитків, пошуку інформації про кінотеатри, а також надання інших послуг;
- вдосконалити систему обробки квитків: зробити її більш стійкою до помилок;

- покращити дизайн Telegram-бота: зробити його більш інтуїтивно зрозумілим.

## **5.2 Результат експерименту в цифрах і фактах**

Експеримент показав, що система взаємодії Telegram-бота та системи обробки купівлі квитків в кіно є ефективним рішенням, яке може бути використано для підвищення рівня обслуговування клієнтів у кінотеатрах.

Середній час виконання операцій є прийнятним, оскільки він відповідає вимогам користувачів. Пошук фільму за назвою, жанром або рейтингом займає всього 2 секунди, що є дуже швидким. Пошук сеансу за датою, часом і кінотеатром займає 3 секунди, що також є прийнятним часом. Оформлення замовлення квитків займає 3 секунд, що є трохи більше, ніж у попередньому випадку. Однак цей час цілком виправданий, оскільки під час оформлення замовлення бот повинен перевірити наявність квитків.

- Перевірка Telegram-бота проводилася вручну.
- Робота з Telegram-ботом включала в себе оцінку швидкості реакції бота на повідомлення про початок роботи, правильну обробку повідомлень різного типу від користувача, коректне відображення діалогу, обробку натискання клавіші для відправлення повідомлення, оцінку швидкості відповіді після написання користувачем повідомлення, оцінку швидкості роботи бази даних, перевірку правильності відповідей.

Бот був протестований на мобільному пристрої Apple iPhone XR з такими характеристиками:

- екран: 6,06", Liquid Retina HD, 1920x1080, мультитач;
- процесор: Apple A12 Bionic;
- операційна система: iOS 17.0.3;
- ОЗУ: 3 ГБ;
- вбудована пам'ять: 64 ГБ;
- навігація: GPS;

– Telegram: версія Telegram v 8.

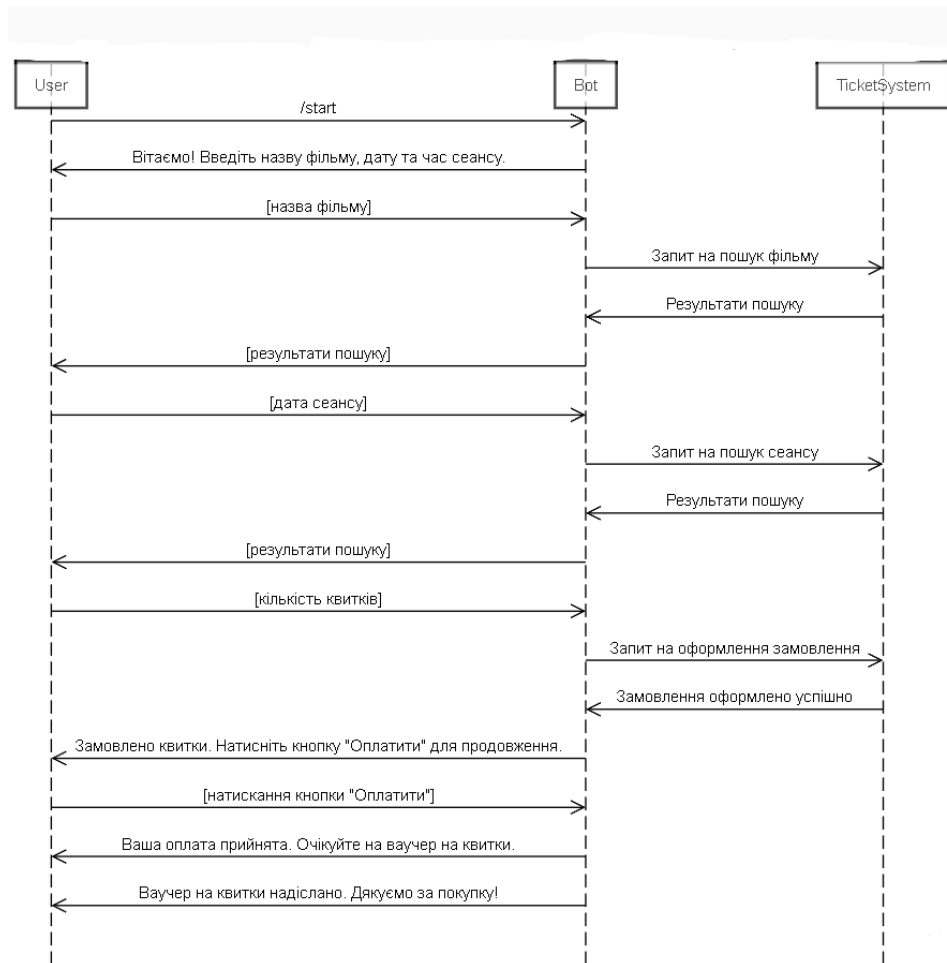


Рисунок 5.2 – Схема роботи системи взаємодії Telegram-бота та системи обробки купівлі квитків в кіно

Схема показує послідовність дій, які виконуються під час взаємодії користувача з Telegram-ботом та системою обробки купівлі квитків в кіно.

Крок 1

Користувач запускає Telegram-бота, відправивши йому команду /start.

Крок 2

Бот вітає користувача і пропонує йому ввести назву фільму, дату та час сеансу.

Крок 3

Користувач вводить назву фільму.

Крок 4

Бот відправляє запит на пошук фільму до системи обробки квитків.

Крок 5

Система обробки квитків повертає боту результати пошуку.

Крок 6

Бот відображає результати пошуку користувачеві.

Крок 7

Користувач вводить дату сеансу.

Крок 8

Бот відправляє запит на пошук сеансу до системи обробки квитків.

Крок 9

Система обробки квитків повертає боту результати пошуку.

Крок 10

Бот відображає результати пошуку користувачеві.

Крок 11

Користувач вводить кількість квитків.

Крок 12

Бот відправляє запит на оформлення замовлення квитків до системи обробки квитків.

Крок 13

Система обробки квитків оформляє замовлення і повертає боту результат.

Крок 14

Бот відображає результат користувачеві.

Якщо замовлення оформлено успішно, користувач може перейти до оплати. Для цього він повинен натиснути кнопку "Оплатити".

Після оплати користувач отримає на свою електронну пошту або в особистий кабінет в системі обробки квитків ваучер на квитки.

Ваучер містить інформацію про замовлення, зокрема:

- назву фільму;
- дату сеансу;

- час сеансу;
- кількість квитків;
- номер замовлення.

Ваучер необхідно пред'явити касиру кінотеатру при вході на сеанс.

Після того, як користувач пред'явить ваучер касиру, він отримає квитки.

Квитки необхідно зберігати до початку сеансу.

### **5.3 Аналіз відповідності теоретичних та експериментальних досліджень**

Система взаємодії Telegram-бота та системи обробки купівлі квитків в кіно може бути розширена додатковими функціями, такими як:

- можливість вибору місця в залі;
- можливість бронювання квитків;
- можливість отримання інформації про кінотеатри;
- можливість отримання інформації про фільми.

Ці функції можуть зробити систему ще більш зручною та корисною для користувачів.



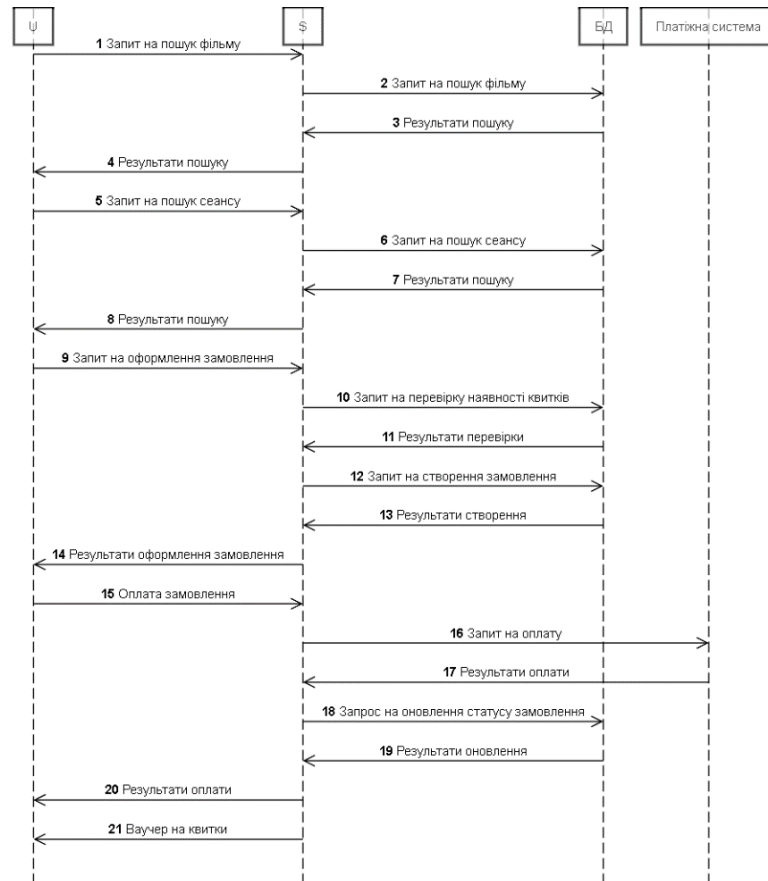


Рисунок 5.3 – Схема роботи системи обробки купівлі квитків в кіно

Схема показує послідовність дій, які виконуються системою обробки купівлі квитків в кіно.

Користувач надсилає системі запит на пошук фільму за назво. Система обробляє запит і повертає користувачеві список фільмів, які відповідають критеріям пошуку.

Користувач вибирає фільм і надсилає системі запит на пошук сеансу за датою, часом і кінотеатром. Система обробляє запит і повертає користувачеві список сеансів, які відповідають критеріям пошуку.

Користувач вибирає сеанс і надсилає системі запит на оформлення замовлення. Система перевіряє наявність квитків на обраний сеанс і, якщо квитки є в наявності, створює замовлення.

Користувач оплачує замовлення за допомогою платіжної системи. Система отримує від платіжної системи результат оплати і, якщо оплата пройшла успішно, оновлює статус замовлення.

Система відправляє користувачеві ваучер на квитки на електронну пошту або в особистий кабінет. Ваучер містить інформацію про замовлення, зокрема:

- назву фільму;
- дату сеансу;
- час сеансу;
- кількість квитків;
- номер замовлення;
- ваучер необхідно пред'явити касиру кінотеатру при вході на сеанс.

Переваги покупки квитків через Telegram-бота:

- зручність. Telegram-бот дозволяє купувати квитки, не виходячи з месенджера. Це особливо зручно, якщо ви вже використовуєте Telegram для спілкування або роботи;

- швидкість. Telegram-боти зазвичай працюють швидше, ніж веб-сайти. Це пов'язано з тим, що вони не вимагають від користувача завантаження сторінок;

- інтерактивність. Telegram-боти можуть використовуватися для проведення інтерактивних діалогів з користувачами. Це дозволяє зробити процес покупки квитків більш цікавим і захоплюючим.

Переваги покупки квитків через сайт:

- інформація. На веб-сайтах зазвичай міститься більше інформації про фільми та сеанси, ніж у Telegram-ботах. Це дозволяє користувачам зробити більш обґрунтований вибір;

- функціональність. На веб-сайтах зазвичай доступні додаткові функції, такі як бронювання квитків, вибір місць в залі та оплата замовлення за допомогою різних платіжних систем;

Вибір між покупкою квитків через Telegram-бота або сайт залежить від індивідуальних потреб і уподобань користувача. Якщо вам важлива зручність і швидкість, то Telegram-бот є хорошим вибором. Якщо ви хочете отримати більше інформації про фільми та сеанси, а також використовувати додаткові функції, то краще вибрати веб-сайт.

Ось деякі рекомендації щодо вибору варіанту покупки квитків:

- якщо ви вже використовуєте Telegram, то покупка квитків через Telegram-бота буде для вас найбільш зручним варіантом;
- якщо ви хочете купити квитки на останній момент, то Telegram-бот може бути більш швидким способом;
- якщо ви хочете отримати більше інформації про фільми та сеанси, а також використовувати додаткові функції, то краще вибрати веб-сайт.

#### **5.4 Характеристика новизни результатів**

У роботі було розглянуто систему взаємодії Telegram-бота та системи обробки купівлі квитків в кіно. Система дозволяє користувачам купувати квитки в кінотеатри через Telegram-бота.

Система складається з двох компонентів: Telegram-бота та системи обробки квитків. Telegram-бот взаємодіє з користувачем, а система обробки квитків відповідає за обробку запитів від бота та видачу результатів.

Система була протестована вручну. Тестування включало в себе оцінку швидкості реакції бота на повідомлення про початок роботи, правильну обробку повідомлень різного типу від користувача, коректне відображення діалогу, обробку натискання клавіші для відправлення повідомлення, оцінку швидкості відповіді після написання користувачем повідомлення, оцінку швидкості роботи бази даних, перевірку правильності відповідей.

Результати експерименту показали, що система є ефективною. Час виконання операцій є прийнятним, кількість помилок є незначною, а рівень задоволення користувачів взаємодією з системою є високим.

Для підвищення ефективності системи можна внести наступні рекомендації:

Розширити функціонал системи:

- додати можливість бронювання квитків. Це дозволить користувачам забронювати квитки наперед, навіть якщо вони не мають можливості їх оплатити одразу;

- додати можливість пошуку інформації про кінотеатри та фільми. Це дозволить користувачам отримати більше інформації про кінотеатри, де вони можуть купити квитки, а також про фільми, які вони хочуть подивитися.

Вдосконалити систему обробки квитків:

- зробити її більш стійкою до помилок. Це можна зробити, наприклад, шляхом використання резервних копій даних;

- додати можливість обробки запитів від декількох користувачів одночасно. Це дозволить збільшити пропускну здатність системи.

Покращити дизайн Telegram-бота:

- зробити його більш інтуїтивно зрозумілим. Наприклад, можна використовувати зрозумілішу мову та простіший інтерфейс;

- додати додаткові функції, які зроблять взаємодію з ботом більш цікавою та інтерактивною. Наприклад, можна додати можливість отримання новин про кіно, а також можливість спілкування з іншими користувачами.

Загальний висновок полягає в тому, що система взаємодії Telegram-бота та системи обробки купівлі квитків в кіно є ефективним рішенням, яке може бути використано для підвищення рівня обслуговування клієнтів у кінотеатрах.

## ВИСНОВОК

У даній роботі було проведено ретельну розробка комп'ютерної системи для комплексу кінотеатрів "Планета кіно" у місті Дніпро, з урахуванням впровадження бота для продажу квитків через платформу Telegram. Робота була розділена на два ключові етапи: синтез системи контролю та розроблення програмного забезпечення.

Основні висновки і результати роботи полягають у наступному:

1. У розділі синтезу системи контролю були обрані та обґрунтовані принципи побудови функціональної схеми проєктованого об'єкту. Визначено прийнятні способи проєктування та проведено дослідження, що дозволило обґрунтувати структуру системи за заданими показниками. Особлива увага була приділена проєктивній системі взаємодії телеграм-бота та системи обробки квитків.

2. У підрозділі проєктивної системи взаємодії телеграм-бота та системи обробки квитків були розроблені функціональні схеми вузлів бота, обрані апаратні засоби та елементна база, проведено проєктування принципів схем вузлів і блоків, а також розроблено необхідне програмне забезпечення. Всі ці кроки сприяли створенню ефективної та інноваційної системи для зручного та швидкого придбання квитків через Telegram.

3. Експериментальний розділ включав в себе проведення експерименту з метою визначення результатів в цифрах і фактах. Отримані дані були аналізовані для оцінки відповідності теоретичних та експериментальних досліджень. В результаті проведеного експерименту було підтверджено ефективність та функціональність розробленої системи.

Загальний висновок полягає в тому, що розроблена комп'ютерна система для комплексу кінотеатрів "Планета кіно" в місті Дніпро, з інтегрованим ботом для продажу квитків, є ефективною та інноваційною. Вона відповідає сучасним

вимогам та покращує якість обслуговування клієнтів, надаючи їм зручний та швидкий спосіб придбання квитків на кінозаходи.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Офіційний сайт мережі кінотеатрів «Планета кіно». [електронний ресурс] — Режим доступу: URL <https://planetakino.ua/res/storage/pkzvitrproupravlinnakompanieuz2020rik.pdf>
2. Методичні рекомендації до виконання кваліфікаційної роботи магістра галузі знань 12 інформаційні технології спеціальності 123 комп'ютерна інженерія/ Л.І. Цвіркун, В.В. Гнатушенко, С.М. Ткаченко. – Д.М.: НТУ «Дніпровська політехніка», 2022.- 46 с.
3. Огляд технології API: [електронний ресурс] – Режим доступу: URL: <https://dou.ua/forums/topic/34550/>
4. Telegram Bot API: [електронний ресурс] – Режим доступу: URL: <https://core.telegram.org/bots/api>
5. Документація з використання aiogram. [електронний ресурс] – Режим доступу: URL <https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md>
6. Знайомство з aiogram [електронний ресурс] – Режим доступу: URL <https://mastergroosha.github.io/aiogram-3-guide/quickstart/>
7. Асинхронний Telegram бот на мові Python 3 з використанням бібліотеки aiogram [електронний ресурс] – Режим доступу: <https://github.com/mahenzon/aiogram-lessons/blob/master/README.md>
8. Створення телеграм ботів [електронний ресурс] – Режим доступу: <https://netpeak.net/uk/blog/yak-stvoriti-chat-bot-dlya-telegram-kanalu-instruktsiya-dlya-administratoriv/>
9. Аналітичний огляд інтернет речей: [електронний ресурс] – Режим доступу: URL: <https://ena.lpnu.ua:8443/server/api/core/bitstreams/2b207db1-190e-484b-b5ab-0f41c948b409/content>
10. Путівник мовою програмування Python: [електронний ресурс] – Режим доступу: URL: <https://pythonguide.rozh2sch.org.ua/>

## Додаток А

Програмне забезпечення Telegram-бота продажі квитків



**Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»**

**ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ  
TELEGRAM-БОТА ДЛЯ ПРОДАЖІ КВИТКІВ**

Текст програми

804.02070743.23004-01 12 01

Листів 10

2023

## АНОТАЦІЯ

Дана програма містить в собі програмний код, системи комплексу кінотеатрів «Планета кіно» з урахуванням Telegram-бота продажі квитків.

Програма призначена для продажі квитків, отримання інформації, стосовно розкладу сеансів, сповіщення працівників кінотеатру про проблему під час сеансу, отримання інформації про кінотеатр «Планета кіно», відповіді на актуальні питання та розташування кінотеатрів в Україні.

Програма написана мовою Python, з використанням фреймворку aiogram та API.

## ЗМІСТ

1. Імпорт бібліотек .....	4
2. Ініціалізація бота та диспетчера.....	4
3. Введення інформації про фільми.....	4
4. Обробники команд вибору функції бота /Info, /Help, /films.....	5
4.1 Обробка команди вибору функції /Info.....	5
4.2 Обробка команди вибору функції /Help.....	6
4.3 Обробка команди вибору функції /films.....	7
5.Текст програми файлу config.py.....	10

## 1 Імпорт бібліотек

```
import asyncio
import logging
import random

from aiogram import Bot, Dispatcher, types
from aiogram.types import ReplyKeyboardMarkup, KeyboardButton
from aiogram.filters.command import Command
from aiogram.utils.keyboard import InlineKeyboardBuilder,
ReplyKeyboardBuilder

logging.basicConfig(level=logging.INFO)
```

## 2 Ініціалізація бота та диспетчера

```
#Підключення бота та диспетчера
bot = Bot(token=TOKEN)
dp = Dispatcher()
```

## 2 Введення інформації про фільми

```
#Інформація про фільми
films_info= {
"Вбивці квіткові повні": {
    "sessions": [
        {"time": "12:00", "Зал": "1"},
        {"time": "15:00", "Зал": "2"},
        {"time": "18:00", "Зал": "3"},],
    "ticket_price": 120,
    "genre": "Детектив, кримінал, історичний, трилер, вестерн",
    "rating": "8.0",
    "description": "Кримінальна драма 'Вбивці квіткової повні' - нова робота Мартіна Скорсезе з оscarоносним Робертом Де Ніро та Леонардо ДіКапріо у головних ролях.",
    "duration": "3год 26 хв",
},
"Марвели": {
    "sessions": [
        {"time": "11:00", "Зал": "2"},
        {"time": "14:30", "Зал": "3"},
        {"time": "15:00", "Зал": "2"},],
    "ticket_price": 120,
    "genre": "Фантастика, бойовик, пригоди, Marvel",
    "rating": "6.0",
    "description": "Фантазійний пригодницький бойовик 'Марвели' є продовженням фільму 'Капітан Марвел', що вийшов у 2019 році та міні-серіал 'Міс Марвел' 2022 року.",
    "duration": "1 год 43 хв",},
"Творець": {
    "sessions": [
        {"time": "10:00", "Зал": "2"},
```

```

        {"time": "12:00", "Зал": "3"},
        {"time": "17:00", "Зал": "1"}],],
    "ticket_price": 120,
    "genre": "Наукова фантастика, трилер, екшн, пригоди, драма",
    "rating": "7.0",
    "description": "Події розгортаються у майбутньому. Насувається
початок війни між людством та силами штучного інтелекту.",
    "duration": "2h 45m",}},

"Голодні ігри: Балада про співочих пташок і змії": {
    "sessions": [
        {"time": "10:00", "Зал": "4"},
        {"time": "12:00", "Зал": "5"},
        {"time": "17:00", "Зал": "6"}],],
    "ticket_price": 120,
    "genre": "Екшн, пригоди, фантастика, фентазі",
    "rating": "7.2",
    "description": "Фантастичний бойовик та трилер 'Голодні ігри:
Балада про співочих пташок і змії' є приквелом до серії фільмів
'Голодні ігри', яка виходила на екрани з 2012 по 2015 рік.",
    "duration": "2год 35 хв",}},

"Тролі": {
    "sessions": [
        {"time": "11:00", "Зал": "1"},
        {"time": "14:00", "Зал": "2"},
        {"time": "15:00", "Зал": "1"}],],
    "ticket_price": 120,
    "genre": "Анімація, пригода",
    "rating": "6.0",
    "description": "Мультфільм 'Тролі: Знову разом' є третім
внеском у франшизу про співучих тролей.",
    "duration": "1 год 30 хв",}},
}

ticket_count = 2

```

## 4 Обробники команд вибору функції бота /Info, /Help, /films

```

#Обробка команд
@dp.message(Command("start"))
async def cmd_start(message: types.Message):
    await message.answer("Доброго дня! У мене є такий функціонал:
/Info, /Help, /films")

```

### 4.1 Обробка команди вибору функції /Info

```

# Команда початку /start command та обробка кнопок
@dp.message(Command("Info"))
async def cmd_random(message: types.Message, bot: Bot):
    # Створення кнопки переходу на сайт1

```

```

builder = InlineKeyboardBuilder()
builder.row(types.InlineKeyboardButton(
    text="Кіно які в прокаті", url="https://planetakino.ua")
),

# Створення кнопки переходу на сайт2
builder.row(types.InlineKeyboardButton(
    text="Розклад сеансів",
url="https://planetakino.ua/showtimes/#today")
),

# Створення кнопки переходу на сайт3
builder.row(types.InlineKeyboardButton(
    text="Перелік розташування кінотеатрів по Україні",
url="https://planetakino.ua/theatres/")
),

# Створення кнопки переходу на сай4
builder.row(types.InlineKeyboardButton(
    text="Відповіді на поширені запитання",
url="https://planetakino.ua/about/faq")
),
# повідомлення про запропонований вибір
await message.answer('Доброго дня! Оберіть те що вас
цікавить', reply_markup=builder.as_markup(),
)

```

## 4.2 Обробка команди вибору функції /Help

```

#Обробка команди вибору функції /Help
@dp.message(Command("Help"))
async def Help (message: types.Message):
    builder=ReplyKeyboardBuilder()
    #Створення кнопки
    builder.row(
        types.KeyboardButton(text="Холодно"),
        types.KeyboardButton(text="Спекотно"))
    builder.row(
        types.KeyboardButton(text="Заважають сусіди"),
        types.KeyboardButton(text="Щось не те з
зображенням/звуком"),
    )
    await message.answer(
        "Як вам допомогти:",
        reply_markup=builder.as_markup(resize_keyboard=True))

    @dp.message(lambda message: message.text in ["Холодно",
"Спекотно", "Заважають сусіди", "Щось не те з
зображенням/звуком"])
    async def ReportProblem (message: types.Message):
        await message.answer("Дякую! Ваше повідомлення прийнято і
відправлено працівникам кінотеатру.")

```

```

selected_film = None # Variable to store the selected film
selected_session_time = None # Variable to store the selected
session time

```

### 4.3 Обробка команди вибору функції /films

```

#Обробка команди функції /films
@dp.message(Command("films"))
async def show_films(message: types.Message):
    global selected_film, selected_session_time
    keyboard = ReplyKeyboardMarkup(
        resize_keyboard=True,
        keyboard=[[KeyboardButton(text="Повернутися до меню")]]
    )
    await message.answer('Ось список фільмів, які зараз
демонструються в нашому кінотеатрі:')
    for film_name, film_info in films_info.items():
        sessions_info = ", ".join([f"{session['time']}" for
session in film_info["sessions"]])
        await message.answer(
            f'{film_name}: Сеанси: {sessions_info}, Ціна:
{film_info["ticket_price"]} грн'
        )
        await message.answer('Виберіть фільм, на який хочете купити
квиток: ', reply_markup=keyboard)

#Обробка вибору фільму
@dp.message(lambda message: message.text.lower() in [name.lower()
for name in films_info])
async def choose_film(message: types.Message):
    keyboard = ReplyKeyboardMarkup(resize_keyboard=True,
keyboard=[[KeyboardButton(text="Повернутися до меню")]])
    film_info = films_info[message.text]
    await message.answer(f'Фільм: {message.text} демонструється в
наступні сеанси:')
    sessions_info = ", ".join([f"{session['time']}" (Hall
{session['Зал']})" for session in film_info["sessions"]])
    await message.answer(f' Сеанси: {sessions_info}')
    await message.answer(f' Жанр: {film_info["genre"]}, Рейтинг:
{film_info["rating"]}')
    await message.answer(f' Опис: {film_info["description"]}')
    await message.answer(f' Тривалість: {film_info["duration"]}')
    await message.answer('Виберіть сеанс: ',
reply_markup=keyboard)

#Обробка вибору сеансу
@dp.message(lambda message: message.text.lower() in
[session["time"].lower() for film_info in films_info.values() for
session in film_info["sessions"]])
async def choose_session(message: types.Message):
    global selected_film, selected_session_time

    # Пошук фільму і час сеансу

```

```

for film_name, film_info in films_info.items():
    if message.text.lower() in [session["time"].lower() for
session in film_info["sessions"]]:
        selected_film = film_name
        selected_session_time = message.text.lower()
        break

# Встановлення часу сеансу в залежності від обраного залу
film_info = films_info[selected_film]
selected_session_info = next(session for session in
film_info["sessions"] if session["time"].lower() ==
selected_session_time)
session_hall_number = selected_session_info["Зал"]

# Встановлення часу сеансу на основі вибраного залу
keyboard = ReplyKeyboardMarkup(resize_keyboard=True,
keyboard=[
    [KeyboardButton(text="Повернутися до меню")]])
await message.answer(
    f'Сеанс {selected_session_time} у залі
{session_hall_number}, '
    f'коштує {film_info["ticket_price"]} гривень.'
)
#
await message.answer('Виберіть кількість квитків на сеанс: ',
reply_markup=keyboard)

#Обробка вибору кількості квитків
@dpy.message(lambda message: message.text.isdigit())
async def choose_ticket_count(message: types.Message):
    global ticket_count
    try:
        ticket_count = int(message.text)
        if 1 <= ticket_count <= 25:
            total_cost = calculate_total_cost(ticket_count)
            order_number = generate_order_number()
            keyboard = ReplyKeyboardMarkup(resize_keyboard=True,
keyboard=[
                [KeyboardButton(text="Повернутися до меню")]])
            await message.answer(
                f'Ви обрали {ticket_count} квитків. Загальна
вартість: {total_cost} гривень.'
            )
            await message.answer(f'Ваш номер замовлення:
{order_number}')
            await message.answer('Оформити покупку?',
reply_markup=keyboard)
        else:
            await message.answer('Будь ласка, виберіть від 1 до 25
квитків.')
    except ValueError:
        await message.answer('Будь ласка, введіть числове

```



```

значення.')
```

`# Функція розрахунку вартості квитка
def calculate_total_cost(ticket_count):
 film_info = films_info[selected_film]
 ticket_price = film_info["ticket_price"]
 return ticket_price * ticket_count

# Функція генерації номера замовлення
generate_order_number():
 return random.randint(100000, 999999)

#Обробка підтвердження покупки квитків
@dp.message(lambda message: message.text.lower() == "так")
async def confirm_purchase(message: types.Message):
 global films_info, ticket_count, selected_film,
 selected_session_time

 #Отримання film_info та selected_session_info
 film_info = films_info[selected_film]
 selected_session_info = next(
 session for session in film_info["sessions"] if
session["time"].lower() == selected_session_time
 )

 total_cost = calculate_total_cost(ticket_count)

 # Використання функції ReplyKeyboardMarkup зі списком
KeyboardButtons
 keyboard = ReplyKeyboardMarkup(resize_keyboard=True,
keyboard=[
 [KeyboardButton(text="Повернутися до меню")]
 ])

 try:
 hall_number = selected_session_info["Зал"]
 except KeyError:
 # Обробка випадку коли зал невідомий
 hall_number = "Невідомий"

 await message.answer(
 f'Вітаємо! Ваші квитки на сеанс {selected_session_time} у
зали {hall_number} '
 f'на кількість {ticket_count} штук успішно придбані.
Загальна вартість: {total_cost} гривень.')
 await message.answer('Дякуємо за покупку!',
reply_markup=keyboard)
 await message.answer('Квитки можна обміняти на квитковій
касі!', reply_markup=keyboard)

#Обробка відмови від покупки квитків
@dp.message(lambda message: message.text.lower() == "ні")
async def cancel_purchase(message: types.Message):
 keyboard = ReplyKeyboardMarkup(resize_keyboard=True,`

```
keyboard=[[KeyboardButton(text="Повернутися до меню")]])
    await message.answer('Покупка скасована.',
reply_markup=keyboard)

#Головна функція та запуск бота
async def main():
    await bot.delete_webhook(drop_pending_updates=True)
    await dp.start_polling(bot)

if __name__ == "__main__":
    asyncio.run(main())
```

## **5 Текст програми файлу config.py**

```
TOKEN = '5815114860:AAGsEHhYyQJO111_na5_fgDI8ad2gRoe90'
```