

ДОДАТОК А

Перелік джерел посилання за науковими напрямами керівника та науковців
кафедри програмної інженерії

1) O. Mazurova, O. Samantsov, O. Topchii and M. Shirokopetleva, A Study of Optimization Models for Creation of Artificial Intelligence for the Computer Game in the Tower Defense Genre, 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), 2020, pp. 491-496, doi: 10.1109/PICST51311.2020.9468057.

33) Shubin, I. , Kozyriev, A. Method for Solving Quantifier Linear Equations for Formation of Optimal Queries to Databases CEUR Workshop Proceedings, 2023, 3396, pp. 449–459

ДОДАТОК Б

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ

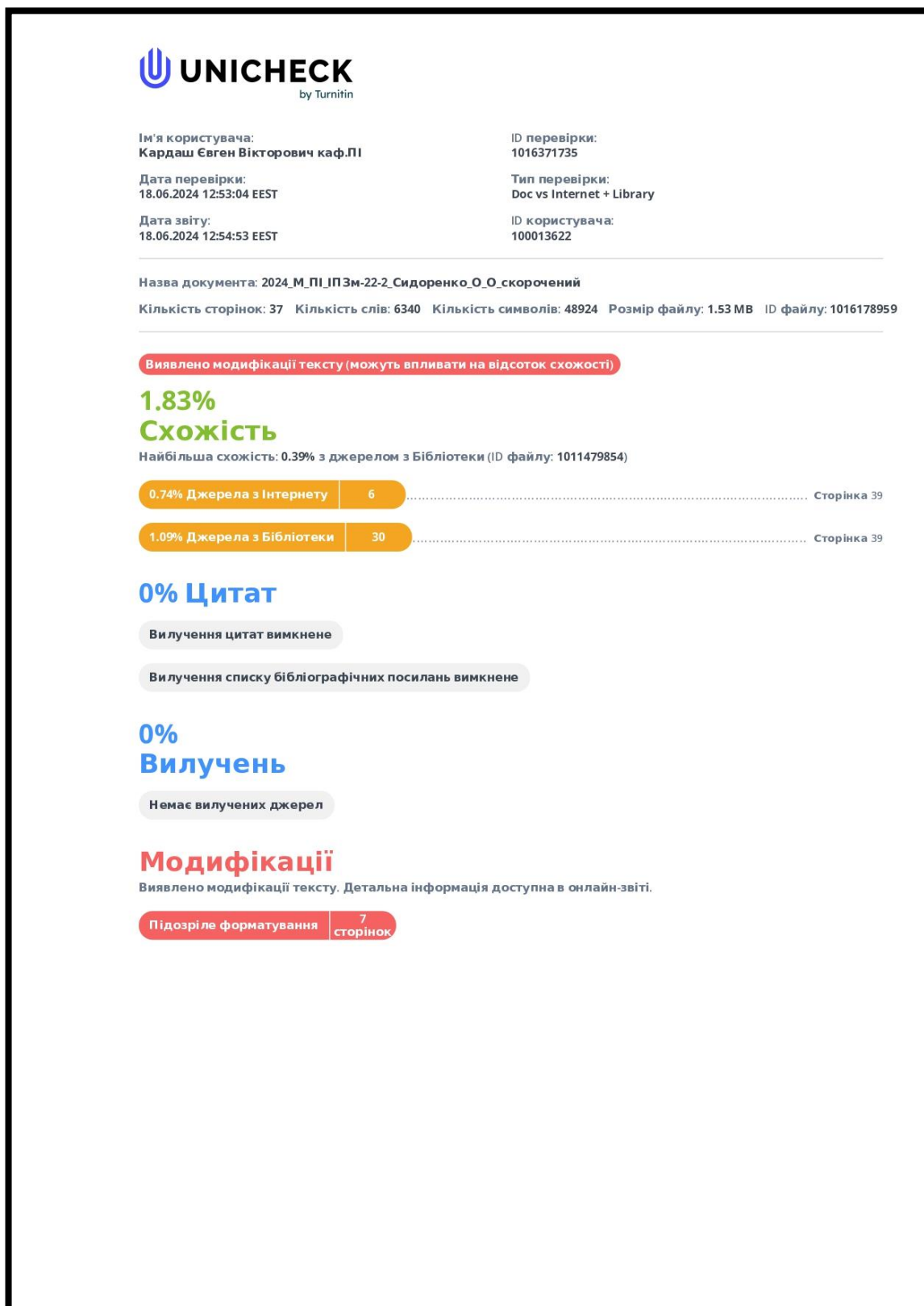


Рисунок Б.1 – Результат перевірки на плагіат

ДОДАТОК В

Апробація у вигляді публікації тез у журналі «International Scientific Unity»



Рисунок В.1 – Титульна сторінка журналу

Theoretical and Practical Aspects of Modern Research

Осипчук Д., Васильцова Н. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ОЦІНЮВАННЯ СКЛАДНОСТІ ЛОКАЛІЗАЦІЇ ІГРОВИХ ПРОГРАМНИХ ПРОДУКТІВ.....	104
Nazarov O., Bobryshev A. ДОСЛІДЖЕННЯ КРИТЕРІЇВ ЯКОСТІ ЗДОБУТТЯ ОСВІТИ У ДИСТАНЦІЙНОМУ ФОРМАТІ НА ОСНОВІ СТАНДАРТІВ РОЗРОБКИ ПРОГРАМНИХ СИСТЕМ.....	109
Sobko O., Mazurets O., Didur V., Chervonchuk I. RECURRENT NEURAL NETWORK MODEL ARCHITECTURE FOR DETECTING A TENDENCY TO ATYPICAL BEHAVIOR OF INDIVIDUALS BY TEXT POSTS.....	113
Сидоренко О.О., Голян В.В. ДОСЛІДЖЕННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ПРИ СТВОРЕННІ ТЕЛЕГРАМ-БОТА ДЛЯ ДОПОМОГИ ПОКУПЦЯМ ПРИ ПОШУКУ ІГРОВИХ ЦІННОСТЕЙ.....	118
Попова А., Васильцова Н. ОЦІНЮВАННЯ УСПІШНОСТІ УПРАВЛІННЯ ЗМІНАМИ В ДОВГОСТРОКОВИХ ІТ-ПРОЄКТАХ.....	120
SECTION: JOURNALISM	
Попович Ю.О. ПОЧАТКИ УКРАЇНСЬКОЇ ПЕРІОДИКИ НА БУКОВИНІ ТА РОЛЬ ГАЗЕТИ «БУКОВИНА» У НАЦІОНАЛЬНОМУ ВІДРОДЖЕННІ УКРАЇНЦІВ КРАЮ.....	125
SECTION: JURISPRUDENCE	
Munteanu R., Baltaga D., Munteanu R. THE EVOLUTION OF METAPHYSICAL AND EPISTEMOLOGICAL IDEALISM: FROM AQUINAS TO MODERN RATIONALISTS.....	128
Бойко В., Котенко С. СУЧАСНІ ПРОБЛЕМИ ЗАХИСТУ АВТОРСЬКИХ ПРАВ В ІНТЕРНЕТІ.....	134
Бочарова Н.В. СТАНОВЛЕННЯ ПРАВОЛЮДИННОГО ПІДХОДУ ДО РЕГУЛЮВАННЯ ІНТЕРНЕТ-ВІДНОСИН В ЄС.....	135

ДОСЛІДЖЕННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ПРИ СТВОРЕННІ ТЕЛЕГРАМ-БОТА ДЛЯ ДОПОМОГИ ПОКУПЦЯМ ПРИ ПОШУКУ ІГРОВИХ ЦІННОСТЕЙ

Сидоренко Олексій Олександрович
здобувач вищої освіти магістерського рівня
oleksii.sydorenko@nure.ua

Голян Віра Володимирівна
кандидат технічних наук, доцент
vira.golan@nure.ua

Кафедра програмної інженерії (ПІ)
Харківський національний університет радіоелектроніки, Україна

Сучасний розвиток технологій створює нові вимоги та можливості для взаємодії людей у віртуальному ігровому середовищі.

RMT (Real Money Trading) означає процес обміну віртуальних активів, таких як ігрові предмети, валюта чи акаунти, на реальні гроші. Це явище стало особливо популярним з розвитком масових багатокористувацьких онлайн-ігор (ММО), де гравці можуть купувати, продавати та обмінювати свої віртуальні здобутки.

RMT відбувається через різні платформи, де гравці можуть виставляти на продаж свої віртуальні активи. Один із прикладів такої платформи — FunPay [1], яка забезпечує безпечний обмін між покупцями та продавцями ігрових цінностей.

FunPay є великою геймерською біржею, де можна придбати та продати різноманітні товари для ігор, такі як зброя, золото, внутрішньоігрова валюта, скіни та інші внутрішньоігрові товари. Основні етапи взаємодії покупця з біржею включають вибір гри, пошук необхідного лоту, обговорення деталей з продавцем, здійснення оплати та підтвердження виконання замовлення. Водночас, процес пошуку та моніторингу пропозицій вручну займає значний час і може бути неефективним.

Одним із ключових завдань при купівлі ігрових цінностей є спрощення процесу пошуку дійсно вигідної або необхідної пропозиції. Telegram-бот, який стане надійним помічником для гравців у пошуку та придбанні цінних ігрових ресурсів на платформі FunPay, є важливим інструментом для вирішення цього питання.

Користувачі FunPay в основному є геймерами, які активно шукають ігрові ресурси. Аудиторія месенджера Telegram включає широкий віковий спектр користувачів, що робить його ідеальним інструментом для сповіщень та взаємодії з користувачами. Кожного дня понад 2,5 мільйона нових користувачів реєструються у Telegram, і на початку 2023 року Telegram перевищив позначку в 800 мільйонів активних користувачів щомісяця [2], що робить цей месенджер популярним серед різних категорій населення.

Theoretical and Practical Aspects of Modern Research

Наразі існує низка ботів для взаємодії користувачів з біржею FunPay, але вони здебільшого орієнтовані на продавців і не задовольняють потреби покупців. Telegram-бот для покупців, що отримує сповіщення про зміни у таблицях пропозицій, є новим і необхідним інструментом для покращення процесу пошуку ігрових цінностей. Telegram-бот повинен мати простий інтерфейс для налаштування пошукових запитів і отримання сповіщень про зміни у пропозиціях. Основні функції включають запуск бота, налаштування пошуку, отримання сповіщень та адміністрування доступу. Це забезпечить ефективну та зручну взаємодію користувачів з ботом.

Для реалізації Telegram-бота можна використати трирівневу архітектуру [3], яка включає:

- рівень представлення (інтерфейс користувача): взаємодія з користувачем через Telegram;
- рівень бізнес-логіки (обробка змін у пропозиціях): обробка змін у таблицях пропозицій та надсилання сповіщень;
- рівень доступу до даних: взаємодія з базою даних.

Ця архітектура дозволяє відокремити логіку користувацького інтерфейсу від бізнес-логіки та обробки даних, що полегшує розширення та підтримку системи.

Для комунікації між компонентами використовується REST API [4], який забезпечує простоту інтеграції та масштабованість. REST використовує стандартні HTTP-методи та протоколи, що робить його простим для використання та розуміння, а також широко підтримуваним у різних системах.

Telegram-бот має забезпечувати:

- простий та зрозумілий інтерфейс для налаштування і отримання сповіщень;
- можливість персоналізації налаштувань для кожного користувача;
- швидку обробку даних та надсилання сповіщень у режимі реального часу;
- захист особистої інформації та безпеку передачі даних;
- ефективну роботу при збільшенні обсягів даних та користувачів;
- логування подій та можливість створення звітів для аналізу роботи системи;
- доступну та зрозумілу документацію до продукту;
- підтримку та можливість вирішення технічних проблем.

Розробка Telegram-бота для сповіщень про зміни у таблицях пропозицій FunPay дозволить значно спростити процес пошуку ігрових цінностей для користувачів. Використання трирівневої архітектури та REST API забезпечить ефективну та безпечну взаємодію з платформою FunPay, що дозволить швидко та зручно отримувати актуальну інформацію про доступні ігрові цінності.

Список використаних джерел

1. FunPay – біржа ігрових цінностей. URL: <https://funpay.com/uk/> (дата звернення: 03.06.2024).
2. Telegram: Contact @durov. URL: <https://t.me/durov/215> (дата звернення: 03.06.2024).
3. Клієнт-серверна архітектура. URL: <https://training.qatestlab.com/blog/technical-articles/client-server-architecture/> (дата звернення: 03.06.2024).
4. Що таке rest api: основні принципи та практики застосування. URL: <https://foxminded.ua/shcho-take-rest-api/> (дата звернення: 03.06.2024).

ОЦІНЮВАННЯ УСПІШНОСТІ УПРАВЛІННЯ ЗМІНАМИ В ДОВГОСТРОКОВИХ ІТ-ПРОЄКТАХ

Попова Анастасія

здобувач магістерського освітнього ступеня
anastasiia.diachenko@nure.ua

Васильцова Наталія

канд. техн. наук, доцент, професор
Кафедра інформаційних управляючих систем
nataliia.vasytsova@nure.ua

Харківський національний університет радіоелектроніки, Україна

На сьогоднішній день ІТ-проекти охоплюють такі основні сфери діяльності: розробку і розвиток програмних застосунків, впровадження інформаційних систем і технологій (ІТ), розгортання ІТ-інфраструктури.

Множину таких проектів можна класифікувати за різними ознаками: за спрямованістю (або сферою діяльності), за масштабом (за обсягами фінансування), за тривалістю реалізації, за складністю реалізації, за призначенням, за характером учасників, за залежністю від вимог до якості, за географічною ознакою тощо [1].

Для довгострокових проектів, які розглядаються в роботі, кожна з цих ознак суттєво впливає на ефективність виконання усіх етапів життєвого циклу проекту.

Такі проекти часто розбиваються на декілька етапів або фаз, кожна з яких має свої цілі, завдання та результати, що дозволяє краще контролювати прогрес і вчасно вносити необхідні корективи.

Але такі проекти мають також й проблеми з узгодженням етапів і фаз при управлінні та оцінюванні результатів цього управління.

Довгострокові проекти часто вимагають комплексного підходу до управління ресурсами, їхнього планування та оптимізації.

Одним із важливих процесів в управлінні довгостроковими ІТ-проектами є управління змінами, які вносяться при реалізації проекту.

Collection of abstracts
XXVI International Scientific and Practical Conference
«Theoretical and Practical Aspects of Modern Research»
June 5-7, 2024
Ottawa, Canada

Organizing committee may not agree with the authors' point of view.
Authors are responsible for the correctness of the papers' text.

Contact details of the organizing committee:

Sole Proprietor Viktoriia Tsiundyk
E-mail: info@isu-conference.com
URL: <https://isu-conference.com/>

Certificate of the subject of the publishing business: ДК №7980 of 03.11.2023.



Рисунок В.6 – Остання сторінка журналу

ДОДАТОК Г

Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимогам ДСТУ 3008: 2015

1

Експертний висновок результатів перевірки кваліфікаційної роботи

студент
(посада)програмної інженерії
(кафедра)ПЗМ-22-2
(група)

Сидоренко О.О.

(прізвище, ім'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.5 Рисунки	
	7.6 Таблиці	
	7.7 Переліки	
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні позначки	
	7.15 Додатки	
Методичні вказівки до виконання кваліфікаційної роботи магістра... ЗАТВЕРДЖЕНО кафедрою ПІ протокол № 5 від 13.11.2023р. 3.2 Оформлення пояснювальної записки згідно з ДСТУ 3008:2015 Звіти у сфері науки і техніки. Структура та правила оформлювання. Шаблон затверджений засіданням кафедри №3 від 16.10.2023.	Рисунок повинен розміщуватися одразу після його згадування у тексті, або на наступній сторінці. Під рисунком повинен бути підпис із словом Рисунок, порядковим номером цього рисунку, через тире з великої літери – назва рисунку та в круглих дужках вказується джерело з якого взятий цей рисунок, або то, що його виконано самостійно.	44

Експерт

(підпис)

Вадим НЕЧВОЛОД

(прізвище, ініціали)

21.06.2024

Рисунок Г.1 – Результат перевірки на нормоконтроль

ДОДАТОК Д

ER-діаграма бази даних

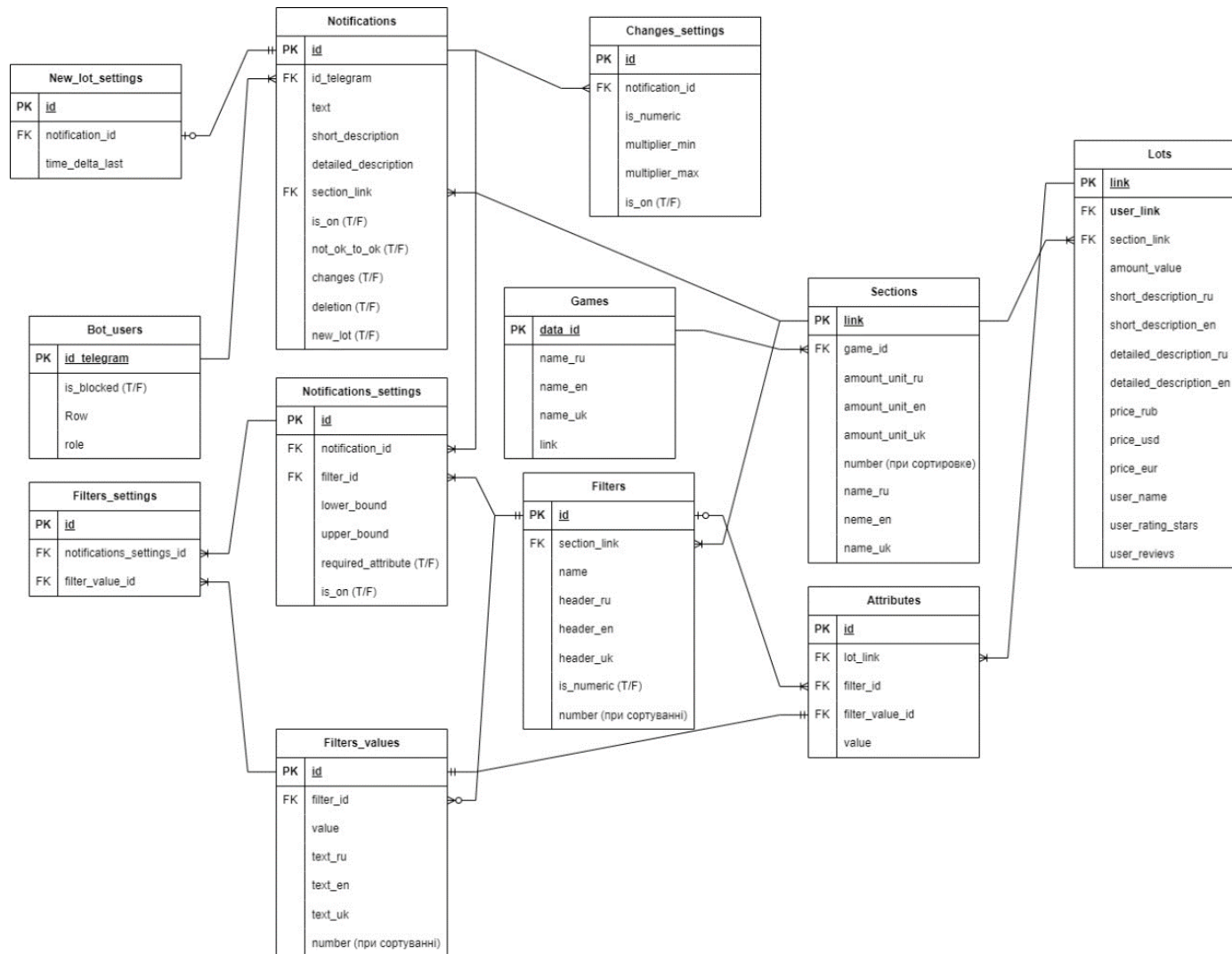


Рисунок Д.1 – ER-діаграма бази даних (рисунок створено самостійно)

ДОДАТОК Е

Фрагменти коду

```
def find_all_games(locale:str = "uk"):
    headers = {'user-agent': UserAgent().random}
    games_dict = {}
    r = requests.get("https://funpay.com/", headers, cookies={"locale": locale})
    soup = BeautifulSoup(r.text, "lxml")
    elements = soup.find("div", class_="promo-games promo-games-all").find("div", class_="promo-game-list")\
        .find_all("div", class_="promo-game-item")

    for element in elements:

        game_titles = element.find_all("div", class_="game-title")
        for game_title in game_titles:
            data_id = game_title["data-id"]
            server = element.find("button", attrs={"data-id": data_id})
            server = f" ({server.text})" if server is not None else ""
            sections = element.find("ul", attrs={"data-id": data_id}).find_all("a")
            sections = [{"link": normalize_link(section.get("href")),
                "name_"+locale: section.text} for section in sections]

            games_dict.update({int(data_id): {"data_id": int(data_id),
                "name_"+locale: game_title.text + server,
                "link": normalize_link(game_title.find("a").get("href")),
                "sections": sections}})

    return games_dict
```

Рисунок Е.1 – Функція пошуку усіх ігор на головній сторінці сайту

```
class Section(Base):
    __tablename__ = "sections"
    link = Column(String, primary_key=True)
    game_id = Column(Integer, ForeignKey("games.id", ondelete="CASCADE"))
    amount_unit_ru = Column(String, default=None)
    amount_unit_en = Column(String, default=None)
    amount_unit_uk = Column(String, default=None)
    name_ru = Column(String, default=None)
    name_en = Column(String, default=None)
    name_uk = Column(String, default=None)
    number = Column(Integer, default=0) для сортування
    game = relationship("Game", back_populates="sections")
    lots = relationship("Lot", back_populates="section",
        cascade="all, delete-orphan")
    filters = relationship("Filter", back_populates="section",
        cascade="all, delete-orphan")
    notifications = relationship("Notification", back_populates="section",
        cascade="all, delete-orphan")
```

Рисунок Е.2 – Клас розділу гри у SQLAlchemy

```

async def update_games_and_sections_tables_with_request(headers: dict = {'user-agent': UserAgent(
    locales = ["ru", "en", "uk"]
    game_data = {}
    section_data = {}

    for locale in locales:
        r = requests.get("https://funpay.com/", headers, cookies={"locale": locale})
        soup = BeautifulSoup(r.text, "lxml")
        elements = soup.find("div", class_="promo-games promo-games-all") \
            .find("div", class_="promo-game-list").find_all("div", class_="promo-game-item")
        for game_element in elements:
            game_titles = game_element.find_all("div", class_="game-title")
            for game_title in game_titles:
                data_id = game_title["data-id"]
                server = game_element.find("button", attrs={"data-id": data_id})
                server = f"({server.text})" if server is not None else ""
                sections = game_element.find("ul", attrs={"data-id": data_id}).find_all("a")
                data_id = int(data_id)

                if data_id not in game_data:
                    game_data[data_id] = {
                        "id": data_id
                    }

                game_data[data_id]["link"] = normalize_link(game_title.find("a").get("href"))
                game_data[data_id]["name_" + locale] = game_title.text + server

                for section in sections:
                    section_link = normalize_link(section.get("href"))

                    if section_link not in section_data:
                        section_data[section_link] = {
                            "link": section_link,
                            "game_id": data_id
                        }

                    section_data[section_link]["name_" + locale] = section.text
            await update_sections_table_from_dict(section_data, game_data)
            await update_games_table_from_dict(game_data)

```

Рисунок Е.3 – Функція оновлення інформації про ігри та розділи у БД

```

class Notification(Base):
    __tablename__ = 'notifications'
    id = Column(Integer, primary_key=True)
    id_telegram = Column(Integer, ForeignKey(BotUser.id_telegram, ondelete="CASCADE"))
    bot_user = relationship("BotUser", back_populates="notifications")
    section_link = Column(String, ForeignKey(Section.link, ondelete="CASCADE"))
    section = relationship("Section", back_populates="notifications")
    text = Column(String, default="+")
    short_description = Column(String, default="-")
    detailed_description = Column(String, default="-")
    is_on = Column(Boolean, default=False)
    not_ok_to_ok = Column(Boolean, default=False)
    changes = Column(Boolean, default=False)
    change_settings = relationship("ChangeSetting", back_populates="notification",
                                   cascade="all, delete-orphan")
    notification_requests = relationship("NotificationRequest", back_populates="notification",
                                        cascade="all, delete-orphan")
    deletion = Column(Boolean, default=False)
    new_lot = Column(Boolean, default=False)
    new_lot_setting = relationship("NewLotSetting", back_populates="notification",
                                   cascade="all, delete-orphan")
    notification_settings = relationship("NotificationSetting", back_populates="notification",
                                        cascade="all, delete-orphan")

```

Рисунок Е.4 – Клас сповіщення у SQLAlchemy

```

def find_info_about_user(soup):
    mini_soup = soup.find("div", class_="media-body")
    name_soup = mini_soup.find("div", class_="media-user-name")
    name = name_soup.text.strip()
    link = soup.find("div", class_="avatar-photo").get("data-href")
    reviews_text = mini_soup.find("div", class_="media-user-reviews").text
    f = filter(str.isdecimal, reviews_text)
    reviews_text = "".join(f)
    if (reviews_text == ""):
        reviews_text = "0"
    reviews = int(reviews_text)
    rating_stars_soup = mini_soup.find("div", class_="rating-stars")
    if (rating_stars_soup != None):
        rating_stars = len(rating_stars_soup.find_all("i", class_="fas"))
    else:
        rating_stars = None
    time_on_site = mini_soup.find("div", class_="media-user-info").text
    info = {}
    info.update({"name": name,
                "link": link,
                "reviews": reviews,
                "rating_stars": rating_stars,
                "time_on_site": time_on_site})
    return info

```

Рисунок Е.5 – Функція пошуку інформації про продавця у об'єкті bs4

```

def select_ch_game_kb (user_id, locale = None):
    if locale is None:
        locale = get_user_locale(user_id)
    column_name = getattr(Game, "name_" + locale)
    r = session.query(column_name).all()
    lst_chars = sorted(set([g[0][0].upper() for g in r]))
    back_button = InlineKeyboardButton(text="⬅", callback_data=f"{user_id}:b4s")
    builder = InlineKeyboardBuilder()

    for ch in lst_chars:
        builder.button(text=ch, callback_data=f"{user_id}:b4s:cr:ch:{ch}")
    builder.adjust(8)
    builder.row(back_button, main_menu_button(user_id=user_id, locale=locale))
    return builder.as_markup()

new *
def select_game_kb (list_of_buttons_tuples, user_id, locale):
    builder = InlineKeyboardBuilder()
    for (i, game_id) in list_of_buttons_tuples:
        builder.button(text=str(i), callback_data=f"{user_id}:b4s:cr:g:{game_id}")
    builder.adjust(5)
    back_button = InlineKeyboardButton(text="⬅", callback_data=f"{user_id}:b4s:cr")
    builder.row(back_button, main_menu_button(user_id=user_id, locale=locale))
    return builder.as_markup()

new *
def select_section_kb (user_id, locale, game_id):
    column_name = getattr(Section, "name_" + locale)
    r = session.query(Section.link, column_name).filter(Section.game_id == game_id).all()
    builder = InlineKeyboardBuilder()
    for (sec_id, name) in r:
        builder.button(text=name, callback_data=f"{user_id}:b4s:cr:s:{sec_id}")
    builder.adjust(2)
    column_name = getattr(Game, "name_" + locale)
    r = session.query(column_name).filter(Game.id == game_id).first()
    ch = r[0].upper()[0]
    back_button = InlineKeyboardButton(text="⬅", callback_data=f"{user_id}:b4s:cr:ch:{ch}")
    builder.row(back_button, main_menu_button(user_id=user_id, locale=locale))
    return builder.as_markup()

```

Рисунок Е.6 – Функції для отримання клавіатур при створенні нової заготовки

ДОДАТОК Ж
Слайди презентації

МІНІСТЕРСТВО
ОСВІТИ І НАУКИ
УКРАЇНИ

ХАРКІВСЬКИЙ
НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ
РАДІОЕЛЕКТРОНІКИ

NURE

Тема роботи: Дослідження методів
прогнозування при створенні
Телеграм-бота для допомоги
покупцям при пошуку ігрових
цінностей

Сидоренко Олексій Олександрович
Науковий керівник: к.т.н.,
доцент Голян Віра Володимирівна

SE
software
engineering

25 червня 2024

1


Рисунок Ж.1 – Слайд 1

Дослідження

Актуальність та стан розвитку галузі

Напрямок дослідження

Об'єкт дослідження



SE
software
engineering

2

Рисунок Ж.2 – Слайд 2

Огляд аналогів

The slide displays three related images. On the left is a screenshot of a terminal window showing a complex ASCII art logo for 'FunPayVertex' with associated metadata including version 'v0.1.12.20' and social media links for GitHub and Telegram. In the center is a blue double-headed vertical arrow with the word 'MANAGER' in blue capital letters to its right. On the right is a screenshot of the 'FunPay Helper' interface, featuring the 'FunPay' logo in a colorful font, the text 'Helper', and a white cartoon character with a red 'FP' on its chest. The 'SE software engineering' logo is located in the bottom left corner of the slide, and a small number '3' is in the bottom right corner.

Рисунок Ж.3 – Слайд 3

Постановка задачі

The slide shows a screenshot of the Roblox Robux marketplace. At the top, it says 'Roblox Robux' and provides a brief description of the marketplace. Below this is a grid of various game passes and items, each with a name and a price in Robux. A 'Choose a server' arrow points to the 'Method of transfer' dropdown menu. Below the dropdown is a table of listings. The 'SE software engineering' logo is in the bottom left corner, and a small number '4' is in the bottom right corner.

Method of transfer	Seller	In stock	Price
Game Pass (5 days)	WareDropShop	∞	0.66 P
Game Pass (5 days)	DINOOSHOP	1	0.38 P

Рисунок Ж.4 – Слайд 4

Методи дослідження

- Аналіз предметної галузі
- Концептуальне проектування
- Прототипування
- Тестування



5

Рисунок Ж.5 – Слайд 5

Інструменти розробки

- Python
- aiogram
- requests, BeautifulSoup4
- SQLAlchemy
- SQLite



6

Рисунок Ж.6 – Слайд 6

Архітектура системи для проведення експериментального дослідження

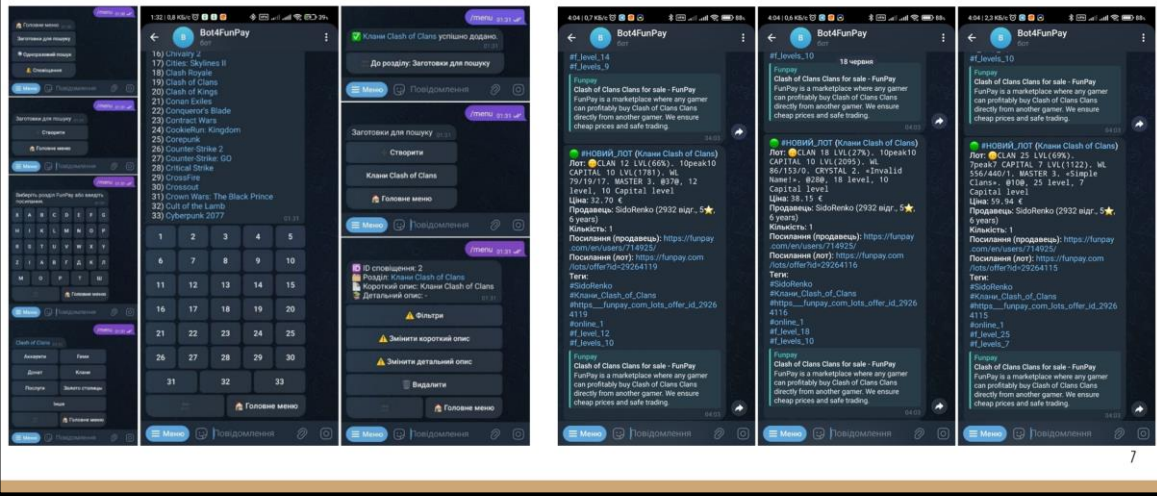


Рисунок Ж.7 – Слайд 7

Етапи розробки

- Планування та аналіз вимог
- Проектування системи
- Реалізація
- Тестування

Рисунок Ж.8 – Слайд 8

Зміст та результати експерименту

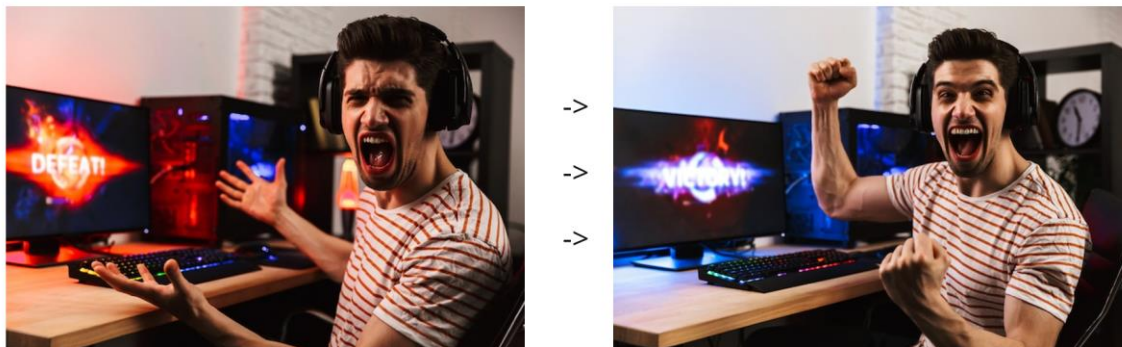


Рисунок Ж.9 – Слайд 9

Аналіз отриманих результатів



Рисунок Ж.10 – Слайд 10

