

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Комп'ютерних наук \_\_\_\_\_  
(повна назва)  
Кафедра \_\_\_\_\_ Системотехніки \_\_\_\_\_  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_  
Розробка інформаційної системи ігрового казино на платформі GTA з  
використанням алгоритмів машинного навчання \_\_\_\_\_  
(тема)

Виконав:  
здобувач \_\_\_\_\_ II \_\_\_\_\_ року навчання,  
групи \_\_\_\_\_ ІТІМ24-2 \_\_\_\_\_  
Мошенський. К. О. \_\_\_\_\_  
(власне ім'я, прізвище)

Спеціальність 122 – Комп'ютерні науки \_\_\_\_\_

(код і повна назва спеціальності)

Тип програми \_\_\_\_\_ освітньо-наукова \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)

Освітня програма Інформаційні технології  
проекткування \_\_\_\_\_

(повна назва освітньої програми)

Керівник \_\_\_\_\_ Доц. Хряпкін О.В. \_\_\_\_\_  
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри \_\_\_\_\_

(підпис)

проф. Гребеннік І.В. \_\_\_\_\_  
(прізвище, ініціали)

2025 р.

## ЗАЯВА

щодо самостійності виконання письмової роботи

Я, Мошенський Костянтин Олегович

(прізвище, ім'я, по батькові)

посада студент

кафедра Системотехніки

заявляю: моя письмова робота на тему Розробка інформаційної системи ігрового казино на платформі GTA з використанням алгоритмів машинного навчання

(назва роботи)

Представлена у екзаменаційну комісію

(спеціалізовану вчену раду, екзаменаційну комісію тощо)

Для публічного захисту, виконана самостійно і в ній не міститься елементів плагіату. Всі запозичені з друкованих та електронних джерел, а також із раніше виконаних дослідницьких робіт та захищених кандидатських і докторських дисертацій мають відповідні посилання.

Я Ознайомлений (а) з діючим положенням «Про протидію плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску письмової роботи до захисту та застосування дисциплінарних заходів.

Дата 15.12.2025

Підпис



Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Системотехніки

Рівень вищої освіти другий (магістерський)

Спеціальність 122 – Комп'ютерні науки

(код і повна назва)

Тип програми освітньо-наукова

(освітньо-професійна або освітньо-наукова)

Освітня програма Інформаційні технології проектування

(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_ СТ

проф. Гребеннік І.В

« \_\_\_\_\_ » \_\_\_\_\_ 2025 р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Мошенський Костянтин Олегович

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка інформаційної системи ігрового казино на платформі GTA з використанням алгоритмів машинного навчання

затверджена наказом по університету від "24" 11 2025 р. № 1058Ст\_24.11.2025

2. Термін здачі студентом роботи до екзаменаційної комісії 21.12.2025

3. Вихідні дані до проекту Проаналізувати існуючі підходи та алгоритми побудови компонентів інформаційних систем віртуальних казино на платформі GTA, зокрема для гри Блекджек, із використанням методів машинного навчання. Обрати оптимальний набір методів аналізу поведінки гравців (класифікація стилів гри, прогнозування ставок та виявлення аномалій) і програмно реалізувати їх, застосовуючи системний підхід, об'єктно-орієнтований аналіз та методи структурного моделювання даних. Розробити модуль інформаційної системи казино, який забезпечує ведення ігрових сесій Блекджеку, збір і зберігання статистики, аналіз рішень гравця та формування інтелектуальних висновків щодо ризиковості або нетиповості його поведінки. Реалізувати вибрані алгоритми з використанням бази даних і web-інтерфейсу адміністративної панелі для перегляду аналітики та результатів роботи моделі.

4. Перелік питань, що потрібно опрацювати в роботі 4.1 Вступ. 4.2 Аналіз предметної області та існуючих підходів до проектування системи. 4.3 Дослідження існуючих підходів, методів. 4.4 Постановка задачі. 4.5 Вибір оптимального методу. 4.6 Розробка вимог до системи. 4.7 Опис прийнятих проектних рішень. 4.8 Висновки 4.9 Перелік посилань. 4.10 Додатки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) \_\_\_\_\_

5.1 Контекстна діаграма IDEF0 інформаційної системи (1 аркуш формату А4). 5.2 IDEF0- діаграма декомпозиції (3 аркуші формату А4). 5.3 Модель даних. (3 аркуші формату А4). 5.6 Реалізована версія інтерфейсу. (6 аркушів формату А4). 5.7 Порівняння метрик (5 аркушів формату А4)

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		(підпис)	(дата)
<i>Аналіз предметної області.</i>	<i>доц. Хряпкін О.В</i>		
<i>Опис прийнятих проектних рішень</i>	<i>доц. Хряпкін О.В</i>		

### КАЛЕНДАРНИЙ ПЛАН

Пор. №	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1.	<i>Отримання завдання кваліфікаційної роботи</i>	13.10.2025	
2.	<i>Ознайомлення з інформаційними технологіями</i>	14.10.2025 – 17.10.2025	
3.	<i>Вивчення предметної області</i>	17.10.2025 – 23.10.2025	
4.	<i>Підбір та вивчення джерел інформації з теми розробки</i>	23.10.2025 – 04.11.2025	
5.	<i>Огляд існуючих підходів технологій та методів у предметній області</i>	04.11.2025 – 05.11.2025	
6.	<i>Постановка задачі розробки та дослідження</i>	23.11.2025	
9.	<i>Вибір оптимального методу</i>	05.11.2025	
11.	<i>Розробка інформаційної системи</i>	12.11.2025 – 03.12.2025	
12.	<i>Планування обчислювальних експериментів</i>	за 5 днів	
13.	<i>Представлення на рецензування</i>	за 3 дні	
	<i>Представлення кваліфікаційної роботи в ДЕК</i>	за 2 дні	

Дата видачі завдання \_\_\_\_\_ 20\_\_ р.

Студент

Мошенський К. О.

(підпис)

Керівник роботи

Доц. Хряпкін О.В

(підпис)

## РЕФЕРАТ

Кваліфікаційна робота: 103 стор., 24 рис., 6 табл., 2 додатки, 25 джерел.

Графічний матеріал кваліфікаційної роботи містить 15 аркушів.

БАЗА ДАНИХ, СЕРВЕРНА ЧАСТИНА, КЛІЄНТСЬКА ЧАСТИНА, GTA,  
RAGE, MYSQL, WEB-SITE

Об'єкт дослідження – процес реалізації діяльності ігрового казино на платформі GTA з використанням алгоритмів машинного навчання..

Предмет дослідження – інформаційні технології та програмні методи створення клієнтської і серверної частин інформаційної системи для реалізації діяльності ігрового казино на платформі GTA.

Мета досліджень – розробка клієнтської та серверної частин інформаційної системи ігрового казино на платформі GTA.

Методи досліджень включають системний підхід, методи структурного аналізу та моделювання реляційних баз даних.

Результати роботи – розроблене програмне забезпечення інформаційної системи ігрового казино на платформі GTA з використанням алгоритмів машинного навчання.

Галузь застосування – включає в себе різноманітні аспекти, спрямовані на розвиток галузі розваг та ігор у віртуальному середовищі.

## ABSTRACT

Qualifying work: 103 p., 24 pic., 4 tables, 2 sources, 25 applications.

Graphic material attestation work contains 15 poster.

DATABASE, SERVER SIDE, CLIENT SIDE, GTA, RAGE, MYSQL, WEB SITE

Object of research – the process of implementing the operation of an in-game casino on the GTA platform using machine learning algorithms.

Subject of research – information technologies and software methods for developing the client-side and server-side components of an information system to implement an in-game casino on the GTA platform.

Purpose of the research – the development of the client-side and server-side components of an information system for an in-game casino on the GTA platform.

Research methods include a systems approach, methods of structural analysis, and modeling of relational databases.

Results of the work – developed software of an information system for an in-game casino on the GTA platform using machine learning algorithms.

Field of application – covers various aspects aimed at the development of the entertainment and gaming industry in a virtual environment.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧУВАНЬ І ТЕРМІНІВ .....	10
ВСТУП.....	11
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПІДХОДІВ ДО ПРОЕКТУВАННЯ СИСТЕМИ.....	14
1.1 Аналіз предметної області.....	14
1.3 Аналіз реалізованих систем ігрового казино .....	17
1.4 Постановка задачі.....	22
2 ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПІДХОДІВ, МЕТОДІВ .....	24
2.1 Типи інтелектуальних систем у казино GTA та принципи їх роботи .....	24
2.2.1 Математичні моделі та алгоритми .....	27
2.3 Метрики оцінки ефективності систем .....	29
2.4 Дослідження варіантів машинного навчання.....	32
3 ВИБІР ОПТИМАЛЬНОГО МЕТОДУ .....	35
3.1 Переваги та обмеження існуючих підходів у системах аналізу гри блекджек.....	35
3.2 Тенденції розвитку та перспективи досліджень .....	39
4 РОЗРОБКА ВИМОГ ДО СИСТЕМИ.....	41
4.1 Розробка системних вимог до інформаційної системи.....	41
4.2 Визначення функціональних вимог до системи .....	42
4.2.1 Розробка та опис діаграми IDF0 .....	42
5 ОПИС ПРИЙНЯТИХ ПРОЕКТНИХ РІШЕНЬ.....	58
5.1 Опис архітектури розробленої системи «Реалізація діяльності ігрового казино на платформі GTA» .....	58
5.2 Обґрунтування вибору мов програмування.....	60
5.3 Обґрунтування вибору СУБД .....	62
5.4 Створення бази даних системи «Реалізація діяльності ігрового казино на платформі GTA».....	67
5.6 Розробка інтерфейсу клієнтської частини.....	71
5.7 Тестування розробленого програмного забезпечення .....	74
5.8 Інтеграція ML-модуля в систему.....	78
5.9 Передача даних із MySQL у ML.NET .....	80
5.10 Виявлення аномалій у реальному часі.....	80

5.11 Сповідання адміністратора .....	82
6 ПРОВЕДЕННЯ ЕКСПЕРИМЕНТІВ ТА АНАЛІЗ РЕЗУЛЬТАТІВ .....	84
6.3 Розподіл аномалій за індексом Anomaly Score (ML-модель) та порогоми правил .....	88
ВИСНОВОК .....	99
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	101
Додаток А .....	104
Додаток Б .....	121
Б.1 ПРИЗНАЧЕННЯ ТА УМОВИ СИСТЕМИ .....	123
Б.1.1 Призначення інформаційної системи ігрового казино .....	123
Б.1.2 Умови застосування інформаційної системи .....	124
Б.2 ПІДГОТОВКА ДО РОБОТИ .....	125
Б.2.1 Розгортання інформаційної системи ігрового казино .....	125
Б.2.2 Запуск системи та перевірка її працездатності .....	125
Б.3 ОПИС ОПЕРАЦІЙ .....	127
Б.3.1 Операції для неавторизованого користувача .....	127
Б.3.2 авторизованого користувача .....	127

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,  
СКОРОЧУВАНЬ І ТЕРМІНІВ

- ІС – інформаційна система;
- БД – база даних;
- СУБД – система управління базами даних;
- ERD – Entity Relationship Diagram (модель даних, яка дозволяє описувати концептуальні схеми предметної області);
- SQL – Structured Query Language (мова структурованих запитів);
- URL – Uniform Resource Locator (уніфіковане посилання на інформаційний ресурс);
- ML – Machine Learning (машинне навчання);
- AI – Artificial Intelligence (штучний інтелект);
- NN – Neural Network (нейронна мережа);

## ВСТУП

У відповідь на швидкий розвиток інформаційних технологій та поширення віртуальних ігор, ігрові казино на платформі Grand Theft Auto (GTA) займають особливе місце у сучасному світі геймінгу. Високий рівень занурення, реалістичність і свобода взаємодії з ігровим світом сприяють появі великої кількості гравців, які прагнуть отримувати нові форми досвіду та додаткові можливості для розваг. Однією з таких можливостей є віртуальні казино, що дозволяють поєднати азарт, динаміку та економічну активність у межах інтерактивного середовища [1] [16].

Інформаційні системи, інтегровані у віртуальні казино GTA, відіграють ключову роль у забезпеченні надійності, безпеки та зручності для користувачів. Вони охоплюють широкий спектр функцій – від керування внутрішньою валютою та фішками до реєстрації гравців, контролю чесності ігрових процесів, оброблення даних та автоматичного управління ігровою логікою. Такі системи забезпечують стабільну роботу віртуального казино, мінімізують ризик технічних збоїв і створюють прозорі механізми взаємодії між гравцем та ігровим середовищем.

Попри широкі можливості платформи GTA, у базовій грі відсутня можливість повноцінної гри в казино, що обмежує потенціал розробників серверів і знижує рівень інноваційності доступних ігрових сценаріїв. Саме тому створення інтегрованої системи ігрового казино, що враховує механіки азартних ігор, поведінкові моделі гравців та сучасні технології аналізу даних, є актуальною задачею.

Зростання популярності онлайн-казино та подібних азартних платформ зумовлене низкою факторів:

- можливість вигравати ігрові фішки або внутрішню валюту, яку можна використовувати у грі, забезпечує додаткову мотивацію та формує елемент прогресії;

– доступність казино з будь-якого пристрою з Інтернет-підключенням робить процес гри зручним і масовим;

– розвиток графічних технологій та інтерактивних механік дозволяє створювати реалістичні геймінг-середовища, у яких гравці можуть відчувати атмосферу справжнього казино, не залишаючи власне робоче місце чи дім.

Актуальність роботи зумовлена необхідністю створення гнучкої та масштабованої інформаційної системи казино, яка може бути інтегрована у GTA-сервери та забезпечувати повний цикл ігрових процесів – від генерації ігрових подій до обробки статистики й управління ставками. Такі системи дозволяють підвищити якість ігрового досвіду, розширити ігрову економіку та збільшити залученість користувачів. [2][3][17]

Об'єктом дослідження є процес функціонування інформаційних систем у віртуальних казино на платформі GTA.

Предмет дослідження охоплює методи побудови та реалізації компонентів інформаційної системи казино, включаючи алгоритми керування ставками, збереження даних, забезпечення чесності гри та автоматизовану обробку результатів.

Мета дослідження полягає у розробці та аналізі інформаційної системи для ігрового казино в середовищі GTA з використанням сучасних алгоритмів і підходів, включно з машинним навчанням, з метою підвищення реалізму, ефективності та безпеки ігрового процесу.

Для досягнення поставленої мети необхідно розв'язати такі завдання:

– виконати аналіз сучасних підходів до реалізації геймінгових інформаційних систем, зокрема у сфері азартних ігор;

– сформувані математичні та логічні моделі ігрових процесів, включно з генерацією випадкових чисел, визначенням виграшів та керуванням балансом;

– дослідити можливості застосування алгоритмів машинного навчання для аналізу поведінки гравців, прогнозування активності та адаптації ігрових пропозицій;

– здійснити оцінку ефективності системи за допомогою метрик стабільності, часу відгуку, коректності обчислень та рівня задоволеності користувачів.

Наукова новизна одержаних результатів полягає у застосуванні алгоритмів машинного навчання для розширення функціональності віртуального казино GTA, зокрема для аналізу поведінкових патернів гравців, підвищення чесності ігрового процесу та створення інтелектуальних модулів адаптивної взаємодії. Запропонований підхід дозволяє оптимізувати внутрішню економіку казино, мінімізувати ризики зловживань та підвищити якість користувацького досвіду.

Практичне значення роботи полягає у можливості інтеграції розробленої інформаційної системи у реальні GTA-сервери. Це забезпечує автоматизацію логіки казино, підвищує надійність і безпечність проведення азартних ігор, розширює можливості для розробників серверів та сприяє зростанню активності й залученості гравців. Застосування інтелектуальних алгоритмів також відкриває перспективи подальшого розвитку ігрових сервісів та створення персоналізованих геймплейних сценаріїв.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПІДХОДІВ ДО ПРОЕКТУВАННЯ СИСТЕМИ

## 1.1 Аналіз предметної області

Розвиток ігрових казино на платформі Grand Theft Auto (GTA) стає дедалі актуальнішим у зв'язку зі зростанням популярності цифрових ігор та інтерактивних віртуальних просторів. Сучасні користувачі все частіше обирають онлайн-формати дозвілля, які поєднують елементи соціальної взаємодії, економічної активності та симуляції реальних сценаріїв. GTA, як одна з наймасштабніших ігрових платформ з відкритим світом, широкими можливостями модифікації та високим рівнем свободи дій, створює сприятливі умови для реалізації інтерактивних казино. [5][24]

Інформаційні системи, що забезпечують роботу казино в GTA, виконують ключові функції: керування ігровими балансами, контроль внутрішньої валюти, забезпечення чесності ставок, обробку даних про дії користувачів та підтримку стабільності серверу. З огляду на стрімке зростання багатокористувацьких серверів із власною економікою, надійність таких систем стає основою якісного та безпечного ігрового процесу.

Онлайн-казино в середовищі GTA пропонують користувачам можливості, що нерідко перевищують функціонал традиційних офлайн-закладів. Зокрема, вони дозволяють:

- робити ставки без ризику втрати реальних коштів, використовуючи внутрішньоігрові гроші;
- отримувати доступ до казино з будь-якого пристрою з Інтернетом;
- спостерігати за діями інших гравців та взаємодіяти з ними;
- користуватися сучасними візуальними та ігровими механіками, що створюють ефект повного занурення. [8]

Гравці високо цінують комфорт, швидкість доступу до гри, відсутність фізичних обмежень, а також безпечне зберігання даних. Завдяки цьому онлайн-казино в GTA приваблюють як новачків, так і досвідчених користувачів.

Одним із важливих напрямів дослідження є аналіз поведінки гравців та внутрішніх процесів прийняття рішень в азартних іграх. Інформація про ставки, частоту участі, вибір конкретних ігор та індивідуальні стратегії дозволяє розробникам оптимізувати ігрові механіки, баланс системи та створювати більш реалістичний геймплей, орієнтований на потреби користувачів. Таким чином, казино в GTA слугує ефективною платформою для тестування поведінкових моделей, економічних систем та інтелектуальних алгоритмів.

З урахуванням постійного вдосконалення ігрових технологій та підвищення вимог гравців до стабільності і безпеки сервісів, інформаційні системи казино мають впроваджувати сучасні методи захисту, серед яких:

- шифрування даних і транзакцій;
- багатофакторна аутентифікація;
- поведінковий аналіз для виявлення відхилень;
- моніторинг серверних процесів;
- механізми протидії шахрайству та несанкціонованому втручанню.

Таким чином, сфера ігрових казино в GTA об'єднує підходи геймдизайну, економічного моделювання, інформаційної безпеки та аналітики поведінки користувачів. Вона має значний потенціал для створення інноваційних рішень, що інтегрують геймінг, статистику та інтелектуальні технології.

## 1.2 Потреби та поведінка користувачів ігрових казино GTA

Спектр поведінкових моделей гравців у віртуальних казино GTA формується під впливом багатьох факторів: індивідуальних уподобань, частоти участі в іграх, мотивації та наявного досвіду взаємодії з азартними механіками. На відміну від класичних наземних казино, де ключову роль відіграють реальні фінансові ризики, у GTA користувачі діють здебільшого з інтересу до нових

емоцій, бажання протестувати власні тактики або заробити внутрішню валюту для подальшого прогресу в ігровому світі.

Під час вибору ігрових активностей у казино GTA гравці орієнтуються на низку критеріїв, серед яких:

- розмір ставок та можливі виграші;
- швидкість проведення ігрових раундів;
- наявність елементів випадковості, ризику та азарту;
- якість візуальних ефектів і загальна атмосфера закладу;
- можливість спостерігати за діями інших учасників або змагатися з ними.

Соціальна взаємодія також відіграє значну роль. Для багатьох користувачів важливими є спілкування, колективні ставки, участь у рейтингах та командних активностях. Таким чином, для більшості гравців важливий не лише кінцевий результат, а й сам процес гри, що підсилює рівень залученості та утримання в системі.

Одним із суттєвих викликів при створенні ігрових казино є явище «cold start», яке проявляється у трьох основних випадках:

- коли до системи приєднується новий гравець, поведінка якого ще не має патернів і не може бути передбачена;
- коли з'являється нова гра, механіки якої не мають накопиченої статистики;
- коли система не володіє достатніми даними для початкового балансування економіки казино.

Усунути ці обмеження допомагає застосування алгоритмів машинного навчання, які здатні аналізувати історію ставок, частоту участі, стратегії гравців, а також загальні закономірності їх взаємодії з казино.

Використання інтелектуального аналізу даних забезпечує можливість:

- адаптувати систему під індивідуальний стиль гри користувача;
- визначати потенційні аномалії або підозрілу поведінку;
- пропонувати персоналізовані рекомендації щодо вибору ігор;
- оптимізувати економічні механізми для запобігання зловживанням;
- прогнозувати серверне навантаження та регулювати доступність ігрових столів.

Окрему увагу слід приділити нормативно-правовому регулюванню діяльності компаній, що створюють ігрові продукти з елементами казино. В Україні розробка подібних систем підпадає під дію низки законодавчих актів, серед яких: Цивільний кодекс України, закон «Про авторське право і суміжні права», закон «Про захист персональних даних», закон «Про електронну комерцію», Податковий кодекс та інші документи, що визначають стандарти безпеки, сертифікації та правила обробки даних.

Таким чином, для створення ефективної та безпечної інформаційної системи казино необхідно враховувати як поведінкові особливості гравців, так і вимоги законодавчого середовища. Поєднання збалансованої ігрової механіки, високого рівня безпеки та стабільної економічної моделі є основою для формування сучасного, надійного та привабливого ігрового сервісу.

### 1.3 Аналіз реалізованих систем ігрового казино

Для створення інноваційної системи, що не має прямих аналогів і здатна конкурувати на ринку, необхідно провести ґрунтовний аналіз уже існуючих рішень, визначити їхні переваги та недоліки, а також систематизувати зібрану інформацію. На основі цього формується концепція продукту, який може перевищувати існуючі аналоги за якістю, функціональністю та зручністю використання. Аналіз доступних інформаційних систем ігрових казино на платформі GTA свідчить про широку різноманітність підходів, орієнтованих на залучення та утримання гравців. Така висока конкуренція стимулює розробників регулярно вдосконалювати свої проекти, впроваджувати новий функціонал та поліпшувати взаємодію користувачів із системою.

Особливої уваги заслуговують сервери, у яких робота казино інтегрована у внутрішню економічну модель. У таких проектах розробники часто реалізують власні унікальні механіки: індивідуальні правила ставок, спеціально стилізовані ігрові автомати, тематичні події та розширені можливості взаємодії. Водночас чимало подібних систем мають суттєві обмеження — використання фіксованих

шансів виграшу, відсутність адаптивного аналізу поведінки гравців та недостатня гнучкість у тонкому налаштуванні й балансуванні механік. Це свідчить про значний потенціал для модернізації, зокрема через інтеграцію інтелектуальних алгоритмів оптимізації та систем, здатних підлаштовуватися під активність користувачів.

Іншу групу конкурентів становлять масштабні рольові сервери (RP), де казино є частиною розгалуженої екосистеми дозвілля. На таких серверах, як правило, пропонуються унікальні інтерфейси, стильні візуальні ефекти та бонусні механіки за активну участь у грі. Проте багато таких рішень приділяють основну увагу естетичній та розважальній складовій, натомість питанням математичної точності, прозорості алгоритмів і чесності результатів — значно менше. Саме тому найбільш перспективними є рішення, що передбачають використання сертифікованих генераторів випадкових чисел, статистичної перевірки результатів та захищеного середовища обробки ставок.

Серед прикладів існуючих реалізацій варто виділити Diamond Casino & Resort [2] — одне з найбільших та найвідоміших віртуальних казино у GTA Online. Воно пропонує широкий вибір азартних ігор, включно з покером, блекджеком, рулеткою та різноманітними слотами. Крім того, у цьому казино реалізовано VIP-систему, що забезпечує гравцям додаткові переваги та бонусний функціонал (рис. 1.3.1).

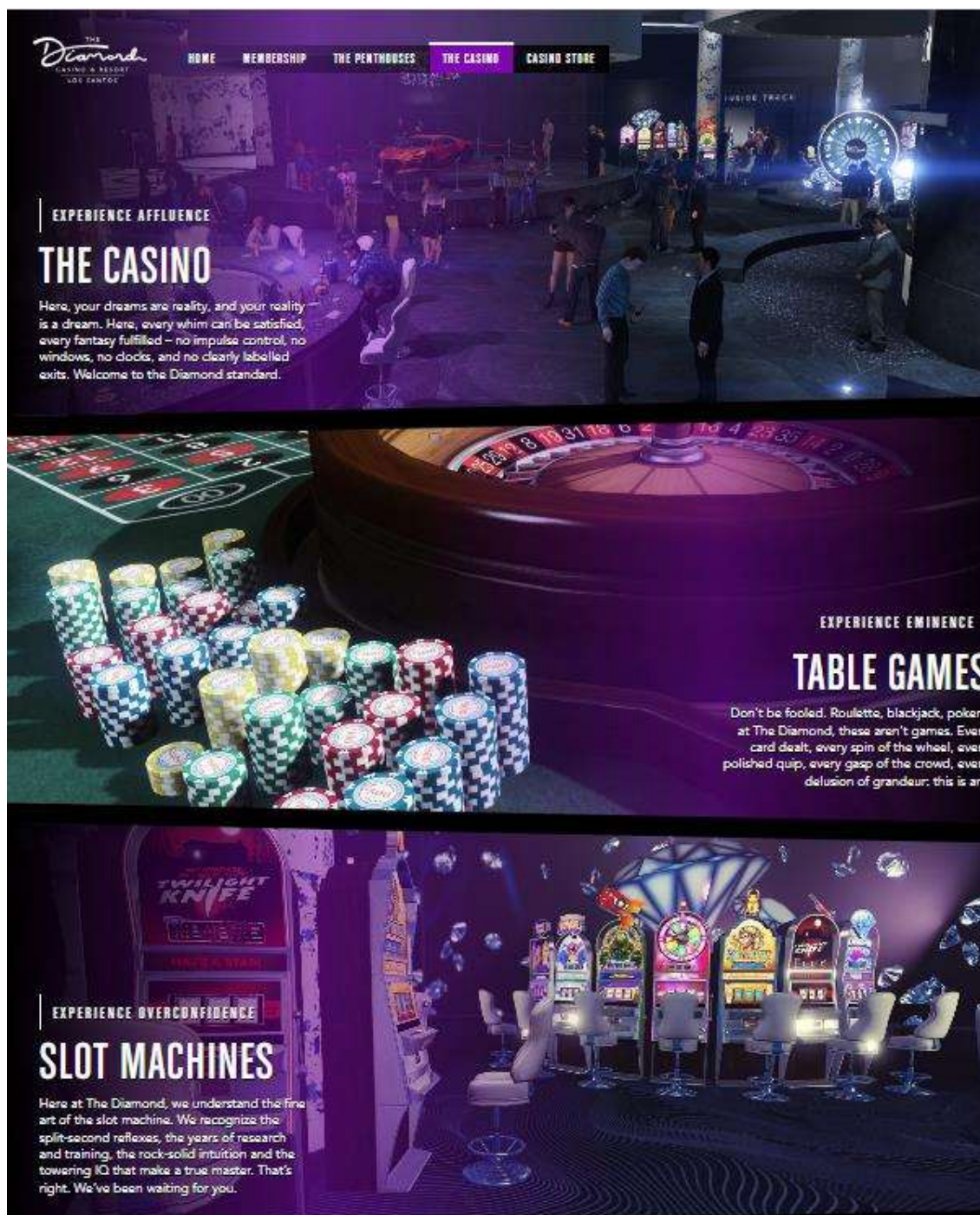


Рисунок 1.3.1 – Інтерфейс ігрового казино Diamond Casino & Resort

Далі треба розглянути The Four Dragons Casino [3]. Це казино, відоме гравцям у всесвіті GTA San Andreas. Воно пропонує широкий вибір азартних ігор, таких як покер, рулетка, карточні ігри та ігрові автомати. Казино також відоме своєю унікальною атмосферою та розвагами для гравців (рис 2.3.2).



Рисунок 1.3.2 – Інтерфейс ігрового казино The Four Dragons Casino

Звичайно, казино є популярною функцією в багатьох іграх, не обмежуючись лише GTA.

В Red Dead Redemption 2 гравці також можуть відвідати казино, яке знаходиться в місті Сент-Дені [4]. Тут їх чекають ігри, такі як покер, блекджек та рулетка, де вони можуть змагатися з іншими NPC або навіть з іншими гравцями у мультиплеєрному режимі (рис 2.3.3).



Рисунок 1.3.3 — Інтерфейс ігрового казино Red Dead Redemption 2

У грі Fallout: New Vegas гравці можуть відвідати різні казино, розташовані в основному місті гри, включаючи відомі заклади, такі як Lucky 38 та Atomic Wrangler [7,8]. Тут вони можуть грати в азартні ігри, такі як слот-машини, рулетка та покер (рис 1.3.4 та рис. 1.3.5).



Рисунок 1.3.4 — Інтерфейс ігрового казино Lucky 38 у грі Fallout: New Vegas

Наявність різноманітних конкурентних рішень активно сприяє розвитку інновацій та покращенню інформаційних систем ігрових казино як на платформі GTA, так і поза її межами. Такі системи змагаються за увагу та прихильність користувачів, постійно розширюючи функціонал і вдосконалюючи механіки, щоб забезпечити максимально якісний геймплей і відповідати потребам гравців. Кожен проєкт намагається виділитися власним підходом до формування ігрового процесу — від нових моделей обробки ставок до персоналізованих елементів та впровадження захищених способів оплати. Завдяки цій конкуренції розробники постійно аналізують тенденції ринку, реагують на зростаючі очікування користувачів та інтегрують сучасні технології для підвищення якості своїх рішень.

## 1.4 Постановка задачі

Завданням кваліфікаційної роботи магістра є дослідження та практичне застосування сучасних методів побудови компонентів інформаційної системи ігрового казино на платформі Grand Theft Auto (GTA) з використанням алгоритмів машинного навчання. Розроблювана система має забезпечувати збір, зберігання та аналіз ігрових даних у модулі Блекджеку, а також формувати інтелектуальні висновки щодо стилю гри користувача, ризиковості його рішень і можливих аномальних дій. Це дозволить підвищити стабільність, безпеку та керованість ігрового процесу, а також забезпечити інструменти аналітики для адміністрації сервера. [6][18]

Для досягнення поставленої мети необхідно вирішити такі підзадачі:

здійснити аналіз сучасних підходів до побудови інформаційних систем віртуальних казино та методів машинного навчання, що застосовуються для аналізу поведінки гравців (класифікація, прогнозування, виявлення аномалій, поведінкова аналітика);

визначити набір ключових параметрів ігрового процесу Блекджеку, що будуть враховуватися в системі (наприклад: розмір ставки, частота рішень hit/stand/double/split, тривалість сесії, виграшні/програшні серії, коливання банкрулу, відхилення від базової стратегії);

обґрунтувати вибір оптимального набору алгоритмів та/або їх комбінації для побудови аналітичного модуля (наприклад: моделі класифікації стилів гри, моделі прогнозування ставок/рішень, методи виявлення підозрілих патернів), які забезпечують найбільш достовірні результати в умовах динамічної зміни поведінки гравців;

спроектувати структуру зберігання даних та модель бази даних для фіксації сесій Блекджеку, ігрових подій, ставок, рішень та результатів, забезпечивши можливість подальшого аналізу й навчання моделей;

створити інтерфейс взаємодії у вигляді веб-сторінок або адміністративної панелі для моніторингу статистики та аналітики, що забезпечуватиме зручність роботи адміністратора (перегляд сесій, звітів, аномалій, профілів гравців);

реалізувати функціонал програмних компонентів системи, який забезпечить обробку ігрових подій у реальному часі, збереження статистики, виконання розрахунків метрик та формування інтелектуальних висновків щодо ризиковості й нетипової поведінки гравців у Блекджеку.

Окрім того, необхідно провести дослідження ефективності обраних алгоритмів машинного навчання шляхом аналізу точності класифікації, відсотку виявлених аномалій, продуктивності моделі та її здатності адаптуватися до змін у поведінці гравців. Це дозволить не лише обґрунтувати доцільність використаного підходу, а й визначити напрямки подальшого вдосконалення системи.

## 2 ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПІДХОДІВ, МЕТОДІВ

### 2.1 Типи інтелектуальних систем у казино GTA та принципи їх роботи

У межах розробки інформаційної системи казино на платформі GTA, яка включає одну ключову гру – блекджек, особливого значення набувають методи інтелектуального аналізу та алгоритми машинного навчання.

Оскільки блекджек є грою з багатьма прихованими закономірностями, стратегічними рішеннями та параметрами, інтелектуальні системи дозволяють:

- аналізувати поведінку гравців;
- адаптувати складність і правила столів;
- формувати персональні рекомендації;
- прогнозувати результати та ризиковість стратегій;
- виявляти аномалії та спроби шахрайства;
- забезпечувати стабільність і чесність ігрової логіки.

Незважаючи на те, що розглядається лише одна гра, її внутрішня структура є досить складною: різні столи блекджека мають власні параметри, які безпосередньо впливають на досвід гравця. До таких параметрів належать: мінімальна ставка, максимальна ставка, кількість колод, регламент дилера (H17/S17), швидкість роздачі карт, можливість удвічі підвищувати ставку, розподіляти руки, а також наявність або відсутність внутрішніх бонусів.

Контентно-орієнтовані системи аналізують характеристики конкретних столів блекджека та підбирають найвідповідніші варіанти під стиль гравця.

Такі підходи ґрунтуються на принципах порівняння об'єктів за їх ознаками та широко описані в класичних роботах з персоналізації контенту.

Для кожного столу формується набір параметрів:

- мінімальна ставка;
- максимальна ставка;
- кількість колод;
- правило дилера (H17 або S17);

- можливість даблзу;
- швидкість раунду;

Щоб визначити схожість між столами, можна застосувати нормалізовану метрику відмінності:

$$D(A, B) = \frac{1}{n} \sum_{i=1}^n \frac{|A_i - B_i|}{\max(A_i, B_i)} \quad (3.1.1)$$

де  $A_i, B_i$  — значення  $i$ -ої характеристики двох столів;

$n$  — кількість ознак

Чим менше значення  $D(A, B)$ , тим більш схожі столи.

Переваги підходу:

- працює навіть тоді, коли система не має достатньо даних про гравців;
- дозволяє точно класифікувати столи за рівнем складності та ризику.

Недолік:

– рекомендації зазвичай дуже схожі на попередні вподобання гравця та можуть зменшувати різноманітність.

Колабораційна фільтрація спрямована на визначення схожих моделей гри між різними користувачами. Цей підхід базується на припущенні, що гравці з подібними стилями поведінки обиратимуть однакові столи або використовуватимуть схожі стратегії ставок.

У випадку блекджека основними ознаками поведінки гравців можуть бути:

- частота хітів та стендів;
- використання подвійної ставки;
- середня ставка на раунд;
- агресивність або консервативність стилю;
- середня тривалість сесії;
- реакція на виграші та програші.

Для оцінки подібності стратегій між двома гравцями можна застосувати коефіцієнт Соренсена–Дайса[3]:

$$S(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|} \quad (3.1.2)$$

де  $X$  — множина рішень або стратегічних дій першого гравця;

$Y$ — множина рішень другого гравця.

Чим ближче значення  $S(X, Y)$  до 1, тим подібніші стратегії.

Переваги підходу:

- дозволяє точно визначити стилі гри;
- ефективний для прогнозування поведінки в майбутніх раундах;
- корисний для рекомендацій “столів за профілем гравця”.

Недолік:

– виникає проблема cold start для нових гравців, які ще не мають достатньої історії рішень.

Гібридні системи поєднують переваги контентного аналізу столів і колабораційної фільтрації стратегій гравців.

Згідно з підходами, такі системи здатні компенсувати недоліки кожного методу, забезпечуючи більш комплексну модель персоналізації.

У блекджеку гібридна система може:

- пропонувати столи на основі параметрів гри;
- підлаштовувати рекомендації під стиль гравця;
- аналізувати ризиковість ставок;
- прогнозувати поведінку у складних ситуаціях;
- визначати відхилення від “нормальної” стратегії (захист від ботових стратегій або маніпуляцій);
- оптимізувати завантаженість столів і баланс роздач.

Інтелектуальні системи дають можливість:

- класифікувати гравців за стилем гри;
- виявляти аномальні рішення;
- прогнозувати середню ставку користувача;
- адаптувати складність столів;
- формувати індивідуальні рекомендації;
- аналізувати ефективність стратегій;
- запобігати шахрайству.

### 2.2.1 Математичні моделі та алгоритми

У попередньому розділі було розглянуто основні типи інтелектуальних систем, що використовуються у віртуальних казино, зокрема контентно-орієнтовані, колабораційні та гібридні. Проте для створення ефективної інформаційної системи необхідно сформулювати математичну модель, яка визначає внутрішню логіку роботи казино, алгоритми оцінки ігрових сценаріїв, аналіз поведінки гравців та прийняття рішень системою. [15]

Інтелектуальна система – це прикладний рівень, який:

- збирає інформацію про ставки та дії гравців;
- аналізує їхні стратегії;
- формує рекомендації чи адаптивні параметри гри;
- визначає ризиковість або аномальність рішень;
- забезпечує чесність і стабільність процесу блекджеку .

Математична модель – це внутрішній механізм, сукупність формул, функцій і алгоритмів, на основі яких система:

- оцінює схожість між гравцями;
- класифікує стилі гри;
- аналізує поведінкові патерни;
- прогнозує майбутні рішення та ставки;
- розраховує ризиковість і очікувану вигоду;
- приймає рішення щодо адаптації складності столів.

Ставки гравця часто є найпоказовішим маркером його поведінкового типу. Для визначення того, наскільки ставка гравця відповідає його типовому рівню, застосовується критерій середньоквадратичного відхилення:

Значення косинусної подібності змінюється від 0 (повна відмінність) до 1 (абсолютна схожість).

Математично цей показник обчислюється за формулою:

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (b_i - \bar{b})^2} \quad (3.2.1)$$

де  $b_i$ — ставка у конкретному раунді,

де  $\bar{b}$ — середня ставка

Якщо ставка значно перевищує критерій середньоквадратичного відхилення, система розцінює це як високий ризик або потенційну аномалію. Щоб краще зрозуміти, як працює ця метрика, розглянемо приклад.

Гравець зазвичай ставить 200–300, але раптом ставить 2000. Алгоритм позначає цю дію як відхилення.

Деякі гравці діють стабільно, інші хаотично. Для оцінювання “рівності стилю” використовується індекс варіативності:

$$V = \frac{\max(b_i) - \min(b_i)}{\bar{b}}, \quad (3.2.2)$$

де  $b_i$ — ставка,

де  $\bar{b}$ — середня ставка

Значення:

- $V < 1$  — стабільний гравець
- $1 \leq V < 3$  — помірно варіативний
- $V \geq 3$  — хаотичний/ризиковий

Наведені моделі дозволяють:

- формувати профілі гравців;
- оцінювати подібність стратегій;
- визначати ризиковість ставок;
- виявляти аномальні рішення;
- прогнозувати майбутні дії;
- оптимізувати роботу столів блекджека.

Ці математичні основи є базою для створення адаптивної інформаційної системи казино, яка здатна аналізувати поведінку користувачів у реальному часі, підвищувати безпеку та забезпечувати чесність ігрового процесу. [9]

### 2.3 Метрики оцінки ефективності систем

Для оцінки якості роботи інформаційної системи казино на платформі GTA, яка аналізує поведінку гравців у блекджеку та формує рекомендації або інтелектуальні висновки, необхідно застосовувати відповідні метрики. Вони дозволяють визначити, наскільки точно система класифікує типи гравців, виявляє аномальні ставки, прогнозує результати рішень або оцінює ризиковість їхньої поведінки.

Оскільки система працює з поведінковими даними, фінансовими характеристиками ставок і математичними моделями ймовірності, застосовуються різні групи метрик, що оцінюють точність, стабільність, ризиковість та прогностичну здатність алгоритмів.

Нижче наведено ключові показники, адаптовані саме під аналіз гри блекджек.

Precision, Recall та F1-score використовуються для ситуацій, коли система повинна визначити, чи є певна дія або гравець аномальним, «ризиковим» або таким, що потребує підвищеної уваги. Наприклад, алгоритм визначає, чи є ставка підозрілою.

$$Precision = \frac{TP}{TP+FP}, \quad (3.3.1)$$

де TP (True Positives) – система правильно позначила ставку як аномальну, FP (False Positives) – система позначила нормальну ставку як аномальну

Precision показує, наскільки добре система «не плутає» нормальні ставки з підозрілими.

Приклад:

Нехай система намагається виявити аномальні ставки (надто великі або нестандартні рішення).

За певний період вона визначила 10 ставок як підозрілі.

З них:

– 7 справді були аномальними  $\rightarrow TP = 7$

– 3 були звичайними, але система їх помилково позначила  $\rightarrow FP = 3$

Тоді:

$$Precision = \frac{7}{7+3} = 0.7, \quad (3.3.2)$$

Інтерпретація:

Точність становить 70 % — система досить добре відокремлює справжні аномалії від нормальних ставок, хоча іноді позначає зайві.

$$Recall = \frac{TP}{TP+FN}, \quad (3.3.3)$$

Де FN (False Negative) – система не помітила аномальну ставку.

Recall демонструє, яку частину справжніх аномалій система виявляє.

Приклад:

У певній ігровій сесії насправді було 12 аномальних ставок.

Система:

– виявила 7 із них  $\rightarrow TP = 7$

– але пропустила 5  $\rightarrow FN = 5$

Тоді:

$$Recall = \frac{7}{7+5} = 0.583, \quad (3.3.4)$$

Інтерпретація:

Recall  $\approx 0.58$ , або 58 % – система знаходить більшість аномалій, але пропускає частину небезпечних ставок.

Це означає, що алгоритм ще потрібно покращувати, щоб підвищити чутливість до підозрілих дій.

F1-score об'єднує в собі попередні показники, створюючи оцінку якості класифікації між Precision і Recall, бо вони оцінюють різні сторони якості системи, для отримання більш правильної оцінки.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}, \quad (3.3.5)$$

Приклад:

Візьмемо попередні значення:

– Precision = 0.7 – Recall = 0.583

Тоді:

$$F1 = 2 * \frac{0.7 * 0.583}{0.7 + 0.583} = 0.636, \quad (3.3.6)$$

Інтерпретація:

$F1 \approx 0.64$ , тобто система має середній баланс між точністю та повнотою:

- вона досить акуратно позначає підозрілі ставки (Precision),
- але не всі аномалії знаходить (Recall),
- і загальний показник відображає обидві сторони.

MAE (Mean Absolute Error) прогнозує розмір ставки або очікувану поведінку гравця, корисно оцінити середню похибку прогнозу:

Формула виглядає так:

$$MAE = \frac{1}{N} \sum_{i=1}^N |p_i - a_i|, \quad (3.3.7)$$

де:  $p_i$  – прогнозована ставка;

$a_i$  – реальна ставка;

Приклад:

Система прогнозує 350, 400 і 300.

Гравець зробив 300, 450, 200.

$$MAE = \frac{|350-300| + |400-450| + |300-200|}{3} = 66.7, \quad (3.3.8)$$

Нижчі значення MAE означають кращу якість прогнозу.

Підсумовуючи, можна зазначити, що жодна окрема метрика не здатна повністю охарактеризувати якість роботи інформаційної системи казино, оскільки кожна з них відображає лише окремий аспект її функціонування. Для отримання об'єктивної оцінки необхідно аналізувати результати комплексно, поєднуючи кілька підходів.

Метрика Precision показує, наскільки система є «акуратною» у визначенні аномальних ставок. Високе значення точності свідчить про те, що система рідко позначає нормальні дії як підозрілі, тобто видає мінімальну кількість хибних спрацьовувань.

Recall характеризує, наскільки система здатна знаходити всі потенційно небезпечні або підозрілі дії. Високий показник Recall означає, що користувач або

оператор системи не пропустить важливих випадків, які можуть свідчити про ризикову поведінку чи спробу маніпуляцій.

Однак висока точність при низькій повноті або навпаки може створювати дисбаланс, тому F1-score використовується як узагальнений критерій, що поєднує обидві метрики. Він дозволяє визначити загальну ефективність класифікації, надаючи збалансовану оцінку здатності системи як точно визначати аномалії, так і знаходити їх у достатній кількості.

Метрика MAE відображає якість прогнозування ставок і поведінкових рішень гравців. Якщо середня похибка є низькою, це означає, що система правильно прогнозує розмір ставок і здатна моделювати майбутні дії користувача з високою точністю. Це важливо для адаптивних алгоритмів, які коригують складність або параметри столів відповідно до рівня гравця.

Таким чином, сукупне використання зазначених метрик дає змогу не лише оцінити ефективність системи виявлення аномалій та прогнозування поведінки гравців, а й забезпечити її стабільну роботу, підвищену безпеку та достовірність аналітичних висновків у процесі гри блекджек на платформі GTA. [4]

## 2.4 Дослідження варіантів машинного навчання

Для реалізації функціоналу аналітики та інтелектуальної обробки ігрових даних у системі віртуального казино необхідно обрати інструмент, який дозволяє ефективно працювати з алгоритмами машинного навчання, зберігаючи при цьому гнучкість, продуктивність та можливість інтеграції з іншими компонентами на платформі .NET. У рамках дослідження проаналізовано популярні підходи та бібліотеки машинного навчання, які можуть бути використані для вирішення задач класифікації, прогнозування та виявлення аномалій.

Серед розглянутих рішень: TensorFlow.NET, Accord.NET, CNTK, а також ML.NET — бібліотека від Microsoft, що забезпечує нативну підтримку .NET-

ядерних додатків без потреби в інтеграції з Python або сторонніми сервісами. Кожен з інструментів має свої сильні сторони, однак саме ML.NET виявився найбільш придатним для поставленої задачі.[2]

Accord.NET — це бібліотека для обробки сигналів і зображень із вбудованими можливостями машинного навчання. Має широкий набір класичних ML-алгоритмів, але є застарілою: розробка практично не ведеться, а інтеграція з новими .NET-версіями обмежена.

CNTK (Microsoft Cognitive Toolkit) — фреймворк для побудови глибоких нейронних мереж від Microsoft. Підтримує потужні моделі, але має складний API і потребує роботи з низьким рівнем, що ускладнює впровадження в типову бізнес-логіку.

TensorFlow.NET — обгортка над TensorFlow, яка дозволяє працювати з TensorFlow у C#. Забезпечує доступ до сучасних DL-алгоритмів, але вимагає детального знання TensorFlow API та суттєвих зусиль при деплої на .NET-сервер.

ML.NET — бібліотека машинного навчання від Microsoft, розроблена спеціально для .NET-екосистеми. Дозволяє працювати з основними задачами ML (класифікація, регресія, аномалії, рекомендації) без виходу за межі C#, має зручний API, автоматичне налаштування моделей (AutoML), інтеграцію з ASP.NET Core і можливість запуску в реальному часі[3].

Переваги ML.NET:

- нативна інтеграція з C#/.NET-проектами;
- підтримка AutoML для підбору найкращих моделей;
- зручна серіалізація та деплой моделей без потреби у зовнішніх інструментах;
- документація та підтримка від Microsoft;
- висока продуктивність і достатня гнучкість для більшості бізнес-задач.

Недоліки інших рішень:

Accord.NET — недостатньо підтримується, не оновлюється;

CNTK — складний поріг входу, не призначений для швидкої розробки;

TensorFlow.NET — надмірна складність і залежність від TensorFlow, що потребує глибоких ML-знань.

У контексті розробки аналітичного модуля для системи віртуального казино на платформі GTA, де важлива швидка розробка, інтеграція з C#, гнучкість та обробка даних у реальному часі — ML.NET є найкращим вибором. Він забезпечує баланс між простотою використання, продуктивністю та можливістю реалізації типових ML-задач без складної інфраструктури та сторонніх залежностей.

### 3 ВИБІР ОПТИМАЛЬНОГО МЕТОДУ

#### 3.1 Переваги та обмеження існуючих підходів у системах аналізу гри блекджек

Інтелектуальні системи, що використовуються у віртуальних казино, зокрема для аналізу гри блекджек, відіграють ключову роль у виявленні підозрілої поведінки, прогнозуванні дій гравців та формуванні адаптивних рекомендацій. Проте жоден із підходів не є універсальним: кожен має власні сильні сторони, обмеження та чутливість до типу даних. Нижче подано порівняння основних методів, що застосовуються у таких системах.

Контентно-орієнтовані моделі у контексті блекджека вважаються характеристики столів та параметри гри[12]:

- мінімальна й максимальна ставка;
- кількість колод;
- регламент дилера (H17/S17);
- швидкість роздачі;
- дозволені дії (double, split тощо).

Контентно-орієнтовані моделі оцінюють схожість столів між собою та пропонують альтернативи на основі параметрів гри.

Переваги:

- не потребують великого масиву даних про гравців – система може працювати навіть із новими або пасивними користувачами;
- легко пояснити логіку: система може однозначно показати, що стіл рекомендовано “через схожу мінімальну ставку” чи “той самий тип правил дилера”;
- добре працюють для оцінки ризиковості столів та підбору оптимального рівня складності.

Недоліки:

- система не враховує реальну поведінку гравця, його стиль і стратегічні особливості;

- можливе «заиклення»: гравцю постійно пропонуються подібні столи, що знижує різноманітність;

- немає адаптації до змін у поведінці – модель реагує лише на характеристики столу, а не на стиль гри користувача.

Колаборативні методи аналізують реальну поведінку гравців у блекджеку:

- частоту рішень (hit, stand, double);

- середні ставки;

- агресивність або обережність;

- схильність до ризику;

- аномальні дії.

Вони ґрунтуються на припущенні, що гравці зі схожими стратегічними патернами будуть діяти подібним чином у майбутньому.

Переваги:

- система не потребує опису столів – працює виключно з поведінковими даними;

- дозволяє виявити приховані закономірності: наприклад, агресивний стиль при низькому банкролі;

- добре масштабується у великих проєктах із тисячами активних гравців.

Недоліки:

- проблема «cold start» – система не може класифікувати нового гравця без історії;

- багато «пробілів» у даних (гравці роблять мало дій, інформації недостатньо);

- чутливість до шумових дій: випадкові або хаотичні рішення спотворюють модель;

- можливість маніпуляцій гравців, які навмисно змінюють стиль, щоб заплутати алгоритм.

Гібридні підходи комбінують обидва підходи:

- параметри столів (контентні дані),
- стиль гри користувача (поведінкові дані).

Це дозволяє отримати більш повну картину та компенсувати недоліки окремих методів.

Переваги:

- зменшують проблему «cold start», оскільки новому гравцю можна рекомендувати столи на основі характеристик, навіть без історії;
- адаптуються до змін поведінки: якщо стиль гравця змінюється, алгоритм це враховує;
- дозволяють одночасно аналізувати і ризиковість столу, і стиль гри;
- створюють більш точні аналітичні висновки.

Недоліки:

- вимагають значно більше обчислювальних ресурсів;
- складні для налаштування та підтримки;
- складніше пояснити: важко визначити, який саме фактор вплинув на рекомендацію;
- необхідність налаштування ваг між моделями, щоб не переоцінювати окремі аспекти гри.

Попри вдосконалення підходів, аналітичні казино-системи в GTA стикаються з низкою типових проблем:

- cold start

Система не може одразу класифікувати нового гравця або передбачити його стиль. Наприклад, новий користувач робить лише 2–3 дії, і даних недостатньо для визначення рівня ризику.

- нестача даних (data sparsity)

Більшість гравців робить мало рішень або мало сесій, що створює розріджені матриці поведінки. Через це алгоритми не можуть визначити схожість між користувачами.

- динамічна зміна стилю гри

Гравці змінюють свій стиль залежно від:

- величини банкролу,
- настрою в сесії,
- програвшої/вигравної серії,
- типу столу.

Це створює проблему адаптації моделі в реальному часі.

- штучні або маніпулятивні стратегії

Гравці можуть навмисно «збивати» алгоритм, здійснюючи нелогічні ставки або різкі зміни стилю гри. Це створює додаткові труднощі для систем виявлення аномалій, особливо якщо вони не враховують динаміку поведінки користувача або мають низьку адаптивність до нових сценаріїв.

Порівняння основних підходів демонструє, що гібридні моделі мають найбільший потенціал для систем аналітики у блекджеку на платформі GTA. Вони поєднують структуровані параметри столів із реальними поведінковими даними гравців, забезпечуючи точніший аналіз у складних ситуаціях. Проте навіть при використанні гібридних систем зберігаються ключові проблеми: нестача даних, динамічна зміна стилю гри та потреба в адаптивному налаштуванні моделей.

У рамках цього дослідження було реалізовано експериментальний підхід на основі ML.NET, що дозволив навчити модель виявлення аномалій на історичних даних ставок. В результаті система отримала можливість присвоювати кожній ставці індекс аномальності (Anomaly Score), що дозволяє виявляти підозрілу поведінку в режимі близькому до реального часу. Такий підхід добре справляється із виявленням нетипових ставок або різких змін у динаміці гри, однак вимагає якісної підготовки даних та періодичного перенавчання моделі.

Для порівняння також було використано експертну (правилу) модель, засновану на фіксованих порогових умовах. Наприклад, якщо ставка перевищує середню більш ніж у 10 разів, або якщо гравець робить кілька різких підвищень за короткий період — така поведінка вважається потенційно аномальною. Цей

підхід є простим у реалізації та не потребує тренування, але значно поступається ML-методам у точності, гнучкості та здатності до масштабування.

У підсумку, саме поєднання адаптивних ML-алгоритмів із базовими правилами може забезпечити найвищий рівень надійності у виявленні шахрайства або нестандартної поведінки. Це також визначає вектор подальших досліджень — оптимізація гібридних систем, адаптація моделей у реальному часі та підвищення стійкості до маніпуляцій зі сторони гравців.

### 3.2 Тенденції розвитку та перспективи досліджень

Сфера інтелектуальних ігрових систем, зокрема систем аналізу поведінки гравців у блекджеку на платформі GTA, активно розвивається під впливом загальних тенденцій штучного інтелекту, машинного навчання та аналітики ігрових даних. З кожним роком зростає потреба у системах, здатних не тільки фіксувати події гри, але й розуміти наміри, стиль і рівень ризику гравця, забезпечуючи більш реалістичну симуляцію казино, балансування ігрового процесу та виявлення аномалій.

Однією з ключових тенденцій розвитку є поведінкова аналітика в реальному часі, яка передбачає обробку дій гравця на рівні окремих рішень (Hit, Stand, Double Down, Split). Сучасні дослідження у сфері гейм-дизайну показують, що персоналізація ігрових сценаріїв підвищує утримання гравців, а застосування аналітичних моделей дозволяє глибше вивчити стилі гри та ризикові патерни.

Іншим напрямом є впровадження систем виявлення аномалій, які дозволяють визначати нетипову поведінку, зокрема:

- підозріло великі ставки;
- нехарактерні рішення у складних ігрових ситуаціях;
- ознаки маніпулятивних стратегій або некоректної гри;
- сценарії, що можуть свідчити про експлойти чи використання ботів.

Також активно розвивається напрям адаптивних систем, які змінюють рекомендації або оцінки ризику залежно від поточного стану гравця. У роботах підкреслюється, що поведінка користувачів у азартних іграх може різко змінюватися під впливом програшів, перемог, зміни банкролу чи темпу гри. Це створює потребу у моделях, які навчаються динамічно, а не лише на історичних даних.

Окремої уваги заслуговує впровадження прогностичних моделей, що передбачають:

- ймовірність ризикової поведінки;
- розмір наступної ставки;
- ймовірність переходу гравця у небезпечний стиль гри;
- очікувану стратегію за даних умов (наприклад, за відхилення від базової стратегії блекджека).

Таким чином, розвиток інтелектуальних систем для казино GTA продовжуватиметься у напрямку глибшої персоналізації, гнучких алгоритмів прогнозування, підвищення точності виявлення аномалій та вдосконалення моделей для реалістичної симуляції поведінки гравців.

## 4 РОЗРОБКА ВИМОГ ДО СИСТЕМИ

### 4.1 Розробка системних вимог до інформаційної системи

Інформаційна система ігрового казино на платформі GTA (зокрема модуль Блекджеку) розроблена для автоматизації проведення ігрових раундів, збору та обробки статистичних даних, а також аналізу поведінкових патернів гравців. Її завдання полягає у формуванні інтелектуальних висновків — наприклад, оцінці рівня ризиковості ставок, виявленні нестандартних або підозрілих дій, визначенні стилю гри користувача. Система враховує індивідуальні особливості гравця, що дає змогу підвищити загальну стабільність, безпеку та контрольованість ігрового процесу, а також надає адміністраторам сервера інструменти для моніторингу та глибинної аналітики. [10]

Виходячи з поставленої задачі, можна виокремити ключові системні вимоги до реалізації такого програмного рішення:

- Серверна частина повинна бути створена на мові C#, оскільки вона забезпечує високу ефективність, надійність і сумісність із серверною логікою GTA, включаючи обробку подій, управління ігровими сесіями, фінансовими транзакціями та статистичними даними.

- Клієнтська частина повинна бути реалізована засобами JavaScript у вигляді UI/CEF-інтерфейсу, що коректно працює всередині клієнта GTA. Інтерфейс має забезпечувати зручну взаємодію з модулем Блекджеку: виконання ставок, вибір дій, відображення карт, результатів і системних повідомлень. [14]

- Комунікація між клієнтом і сервером має здійснюватися за допомогою подієвої моделі (events/callbacks) та/або вбудованих API-механізмів платформи. Така взаємодія гарантує низьку затримку та стабільність роботи в режимі реального часу.

- Система зберігання даних повинна містити базу, у якій фіксуються всі аспекти ігрових процесів: історія сесій, ставки, виконані дії гравців (hit, stand, double, split), результати кожного раунду, динаміка банкрулу та журнали подій.

Ці дані необхідні як для аналітики, так і для навчання алгоритмів машинного навчання.

– Ідентифікація та контроль доступу є обов'язковими елементами системи. Потрібно забезпечити прив'язку кожної дії до конкретного гравця, розмежування ролей (гравець / адміністратор) та захист критичних операцій, таких як фінансові транзакції або адміністрування.

– Аналітичний інтерфейс для адміністрації має надавати розширені можливості перегляду статистики сесій, профілів поведінки користувачів, списків підозрілих дій і результатів роботи інтелектуальних алгоритмів. Такий інструмент може бути реалізовано як у вигляді внутрішнього інтерфейсу, так і веб-панелі.

– Механізми обробки помилок і логування повинні забезпечувати фіксацію збоїв серверної логіки або проблем у взаємодії клієнта й сервера. Це потрібно для мінімізації впливу помилок на ігровий процес і забезпечення швидкої діагностики та відновлення роботи системи.

## 4.2 Визначення функціональних вимог до системи

### 4.2.1 Розробка та опис діаграми IDF0

Для встановлення функціональних вимог до інформаційної системи ігрового казино на платформі GTA було проведено моделювання відповідно до методології IDEF0 [8]. На рис. 5.1 наведено контекстну діаграму основної бізнес-функції.

У межах функції «Реалізація діяльності ігрового казино на платформі GTA» визначено ключові ролі:

«Гравець» — користувач, який успішно авторизувався в системі, має зареєстрований профіль та певний обсяг коштів. Він бере участь в ігрових процесах, виконує ставки та інші пов'язані дії.

Під час аналізу головної бізнес-функції виокремлено такі керуючі дані:

«Документація» — набір регламентів, правил та обмежень, які визначають порядок функціонування казино на платформі GTA.

Також було визначено перелік вхідних даних, що використовуються в межах функції:

«Інформація про гравця» — дані, необхідні для перевірки можливості входу до казино, включно з інформацією про положення гравця та право доступу.

«Дані про баланс рахунку» — показники, які відображають кількість коштів на рахунку гравця та наявність банківської картки.

«Курс для купівлі фішок» — актуальне значення курсу, за яким гравець може придбати фішки; цей курс переважно більший за курс продажу.

«Курс для продажу фішок» — значення курсу, за яким гравець може продати фішки; зазвичай нижчий за курс купівлі.

«Правила рулетки» — набір умов гри, включаючи варіанти ставок, коефіцієнти, часові обмеження та додаткові параметри.

«Інформація про столи рулетки» — відомості щодо локацій столів, мінімальних ставок і технічних характеристик.

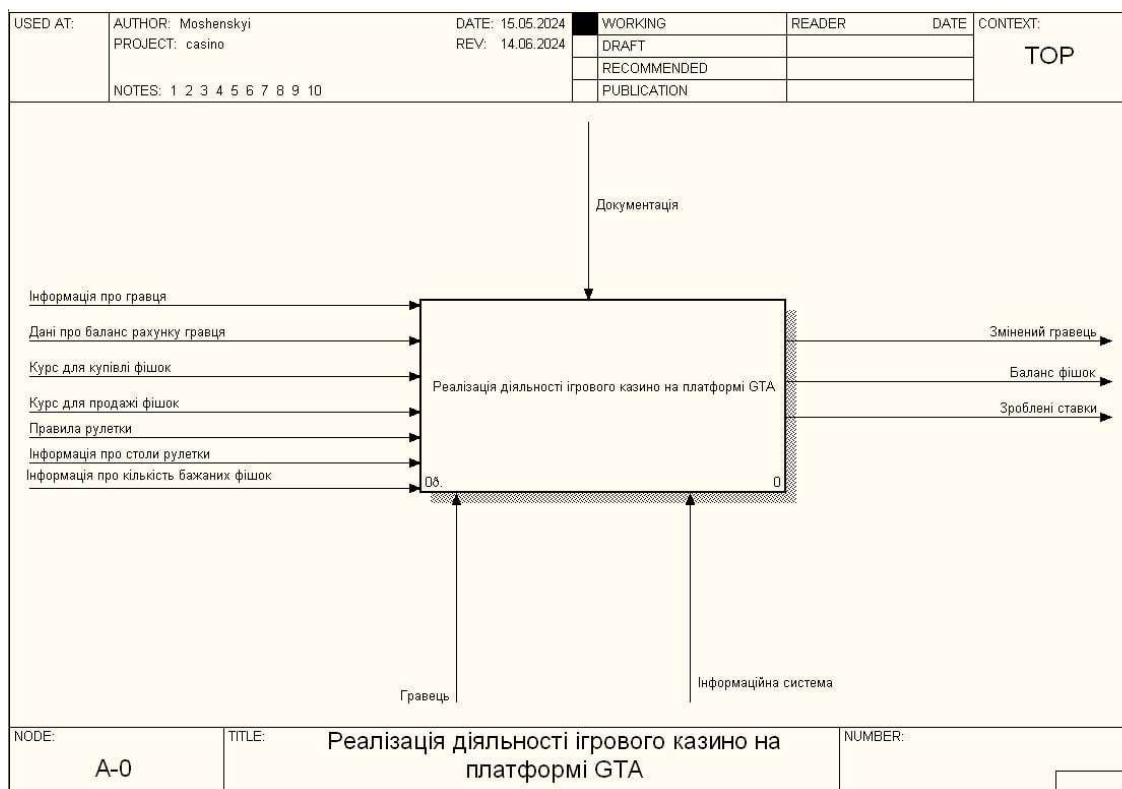


Рисунок 4.1 – IDEF0 Концептуальна діаграма

Під час аналізу функції «Реалізація діяльності ігрового казино на платформі GTA» встановлено такі вихідні потоки:

«Змінений гравець» — інформація, що підтверджує, що гравець увійшов до казино і його стан було змінено відповідно до сценарію ігрової системи.

«Баланс фішок» — оновлене значення кількості фішок після купівлі або продажу.

«Зроблені ставки» — інформація про актуальні ставки гравця, яка коригується залежно від його дій.

Для більш точного опису було створено декомпозицію основної функції (рис. 5.2).

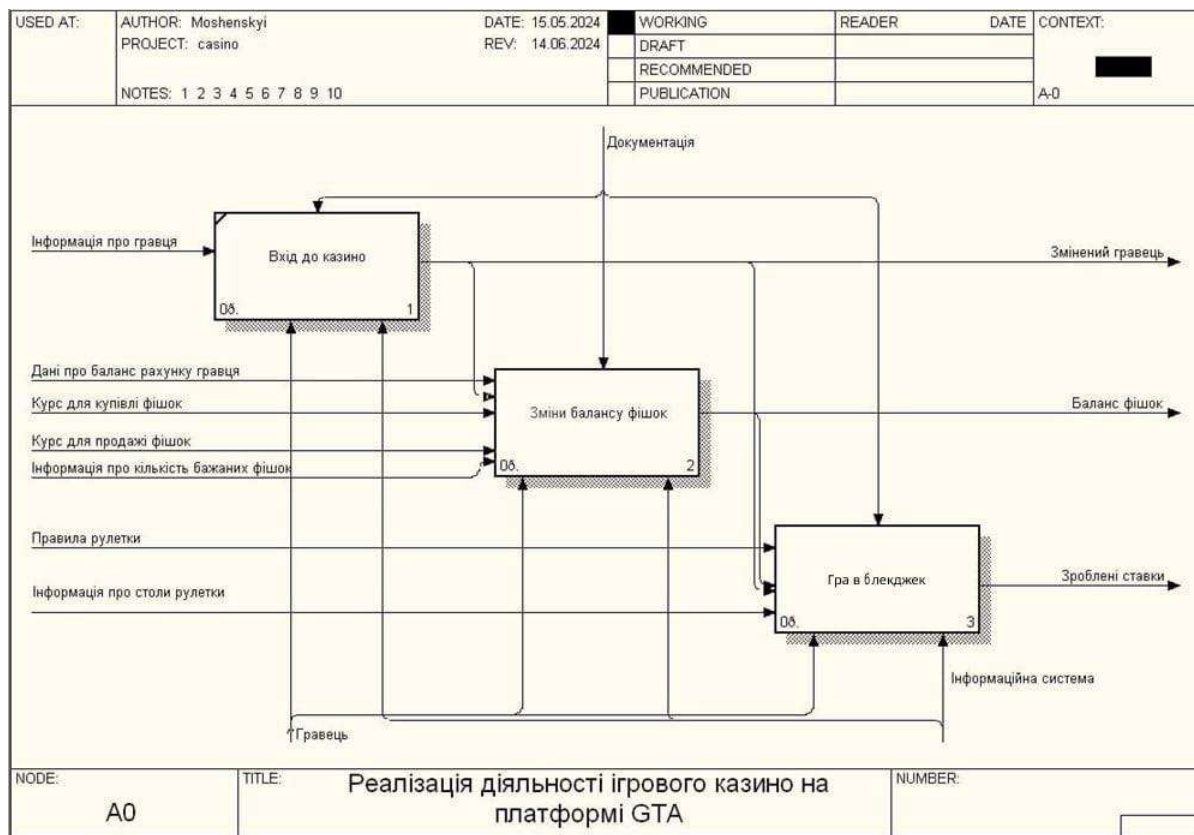


Рисунок 4.2 – IDEF0 Декомпозиція першого рівня

Блок «Вхід до казино» забезпечує реалізацію процесу потрапляння гравця до ігрового казино. Цей блок містить такі дані управління:

«Документація» – нормативні акти, правила та регламенти, що визначають порядок функціонування ігрового казино на платформі GTA.

Бізнес-функція «Вхід до казино» має такі дані механізму:

«Гравець» – користувач, який уже пройшов авторизацію в грі, має створений обліковий запис і певний обсяг коштів. Може користуватися функціоналом казино, робити ставки тощо;

«Інформаційна система» – програмний комплекс ігрового казино, у межах якого відбувається взаємодія гравців для реалізації головної бізнес-функції системи.

Бізнес-функція «Вхід до казино» має такі вхідні дані:

«Інформація про гравця». Ця стрілка містить відомості, що використовуються під час входу до казино, зокрема дані про те, чи перебуває гравець у транспортному засобі та чи має він право потрапити до приміщення казино.

Бізнес-функція «Вхід до казино» має такі вихідні дані:

«Змінений гравець». Ця стрілка містить інформацію про те, що гравець уже увійшов до казино, його розташування змінено відповідно до сценарію, а зовнішній вигляд може бути адаптовано до умов локації.

Блок «Зміна балансу фішок» відповідає за коригування балансу фішок гравця та надає можливість придбання або продажу фішок. Цей блок містить такі дані управління:

«Документація» – сукупність правил і норм, що регламентують процеси купівлі та продажу фішок в ігровому казино на платформі GTA.

Бізнес-функція «Зміна балансу фішок» має такі дані механізму:

«Гравець» – користувач, який пройшов авторизацію, має створений аккаунт і кошти, та може здійснювати ігрові операції, включно зі ставками;

«Інформаційна система» – система ігрового казино, у якій відбувається обробка операцій із фішками та забезпечується реалізація основної бізнес-функції.

Бізнес-функція «Зміна балансу фішок» має такі вхідні дані:

«Дані про баланс рахунку». Ця стрілка містить персональні фінансові дані гравця: кількість його коштів, наявність банківської картки та суму коштів на ній;

«Курс для купівлі фішок». Ця стрілка містить інформацію про актуальний курс купівлі фішок, який змінюється в часі та завжди перевищує курс їх продажу;

«Курс для продажу фішок». Ця стрілка містить дані про поточний курс продажу фішок, що є змінним і завжди нижчим за курс купівлі.

Бізнес-функція «Зміна балансу фішок» має такі вихідні дані:

«Баланс фішок». Ця стрілка містить оновлені відомості про баланс фішок гравця після здійснення операцій купівлі або продажу.

Блок «Гра в рулетку» реалізує можливість участі гравця в грі рулетка, а також надає доступ до перегляду правил цієї гри. Цей блок містить такі дані управління:

«Документація» – закони, правила та внутрішні регламенти, що визначають порядок проведення гри в рулетку в межах ігрового казино на платформі GTA.

Бізнес-функція «Гра в рулетку» має такі дані механізму:

«Гравець» – користувач, який успішно авторизувався в грі, має обліковий запис і необхідні кошти для участі в ігрових процесах, у тому числі для здійснення ставок;

«Інформаційна система» – система ігрового казино, у якій організовано процес взаємодії гравців із рулеткою та реалізовано основну бізнес-функцію системи.

Бізнес-функція «Гра в рулетку» має такі вхідні дані:

«Правила рулетки». Ця стрілка містить повний набір даних про правила гри в рулетку, включаючи варіанти ставок, коефіцієнти виплат, часові обмеження та інші регламентовані параметри;

«Інформація про столи рулетки». Ця стрілка містить дані щодо розташування столів, їхніх характеристик, мінімального розміру ставки та інших параметрів, що впливають на вибір гравцем столу.

Бізнес-функція «Гра в рулетку» має такі вихідні дані:

«Зроблені ставки». Ця стрілка містить інформацію про сформовані гравцем ставки, що оновлюється в разі додавання нових ставок або скасування вже розміщених.

Для детальнішого опрацювання визначених бізнес-функцій системи ігрового казино було проведено подальшу декомпозицію відповідних блоків. Бізнес-функція «Вхід до казино» не потребує додаткової декомпозиції, тоді як

представлена на рис. 5.3 діаграма відображає деталізацію блоку «Зміна балансу фішок».

Блок «Купівля фішок» забезпечує можливість придбання фішок гравцем відповідно до встановленого курсу. Цей блок включає такі дані управління:

«Документація» — нормативна база, що визначає порядок роботи ігрового казино на платформі GTA.

Бізнес-функція «Купівля фішок» містить такі дані механізму:

«Гравець» — користувач, який успішно авторизувався в грі, має створений профіль та необхідні кошти, а також може брати участь у діяльності казино та виконувати ставки;

«Інформаційна система» — програмний комплекс казино, у межах якого відбувається взаємодія гравців та реалізуються основні бізнес-процеси системи.

Бізнес-функція «Купівля фішок» має такі вхідні дані:

«Дані про баланс рахунку» — стрілка містить персональну фінансову інформацію гравця: доступні кошти, наявність банківської картки та суму, що зберігається на ній;

«Курс для купівлі фішок» — стрілка відображає актуальний курс придбання фішок, який є змінним та традиційно перевищує курс їх продажу.

Бізнес-функція «Купівля фішок» має такі вихідні дані:

«Баланс фішок» — стрілка передає оновлений стан балансу фішок гравця після проведення операції купівлі або продажу.

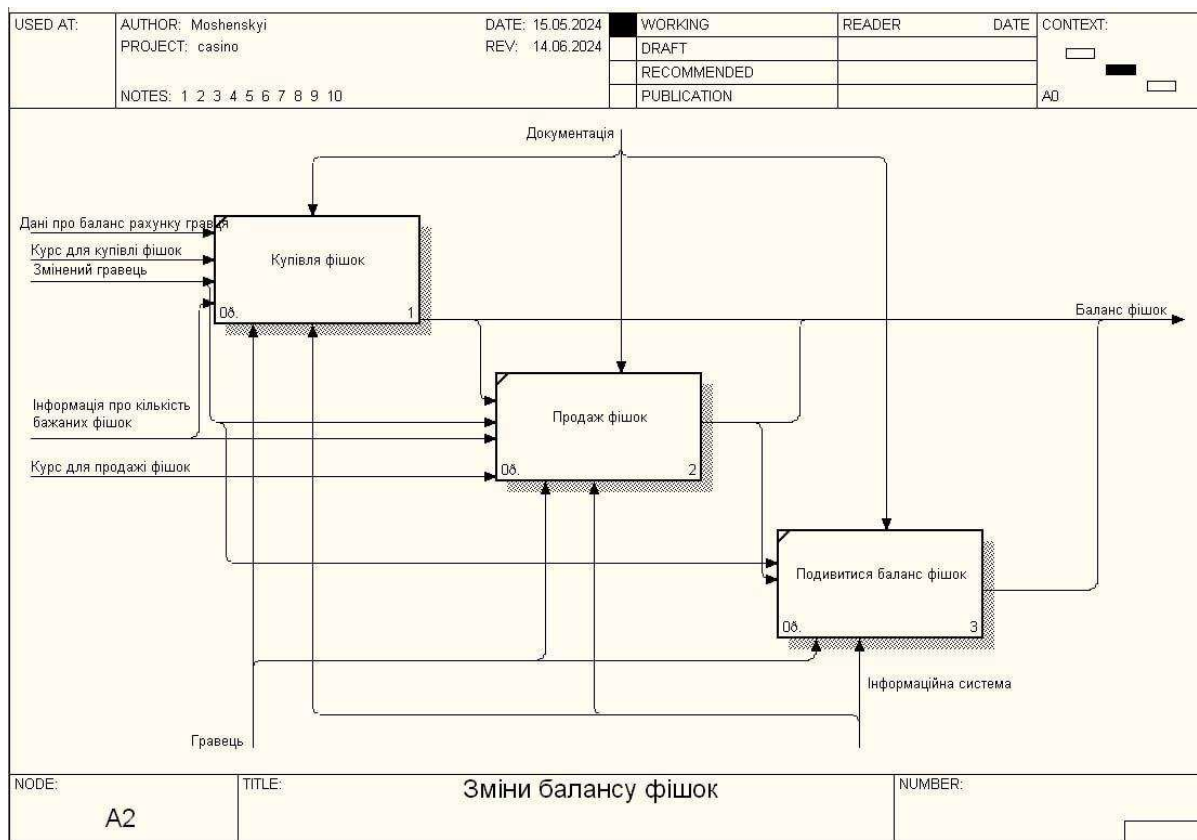


Рисунок 4.3 – Діаграма декомпозиції бізнес-функції «Зміна балансу фішок»

Блок «Продаж фішок» забезпечує можливість реалізації фішок гравцем відповідно до діючого курсу. Цей блок містить такі дані управління:

«Документація» — нормативні правила та вимоги, що визначають порядок здійснення операцій із фішками в межах ігрового казино на платформі GTA.

Бізнес-функція «Продаж фішок» має такі дані механізму:

«Гравець» — користувач, який пройшов авторизацію, має створений обліковий запис та необхідні кошти, і може виконувати ігрові дії;

«Інформаційна система» — система казино, у якій здійснюються всі процеси, пов'язані з реалізацією основних ігрових функцій.

Бізнес-функція «Продаж фішок» має такі входні дані:

«Курс для продажу фішок» — стрілка містить інформацію про змінний курс продажу фішок, який завжди нижчий за курс купівлі.

Бізнес-функція «Продаж фішок» має такі вихідні дані:

«Баланс фішок» — стрілка відображає оновлений баланс фішок після здійснення операції продажу.

Блок «Подивитися баланс фішок» дозволяє гравцеві переглянути кількість доступних фішок. Цей блок містить такі дані управління:

«Документація» — правила та регламенти, що визначають порядок роботи ігрового казино на платформі GTA.

Бізнес-функція «Подивитися баланс фішок» має такі дані механізму:

«Гравець» — авторизований користувач, який має можливість виконувати ігрові взаємодії;

«Інформаційна система» — система казино, через яку гравці здійснюють відповідні дії в межах бізнес-функціоналу.

Бізнес-функція «Подивитися баланс фішок» має такі вхідні дані:

«Баланс фішок» — стрілка містить інформацію про актуальну кількість фішок після проведення операцій купівлі чи продажу.

Бізнес-функція «Подивитися баланс фішок» має такі вихідні дані:

«Баланс фішок» — стрілка повторно відображає оновлену інформацію про баланс гравця.

На рис. 4.4 представлено діаграму декомпозиції бізнес-функції «Гра в блекджек».

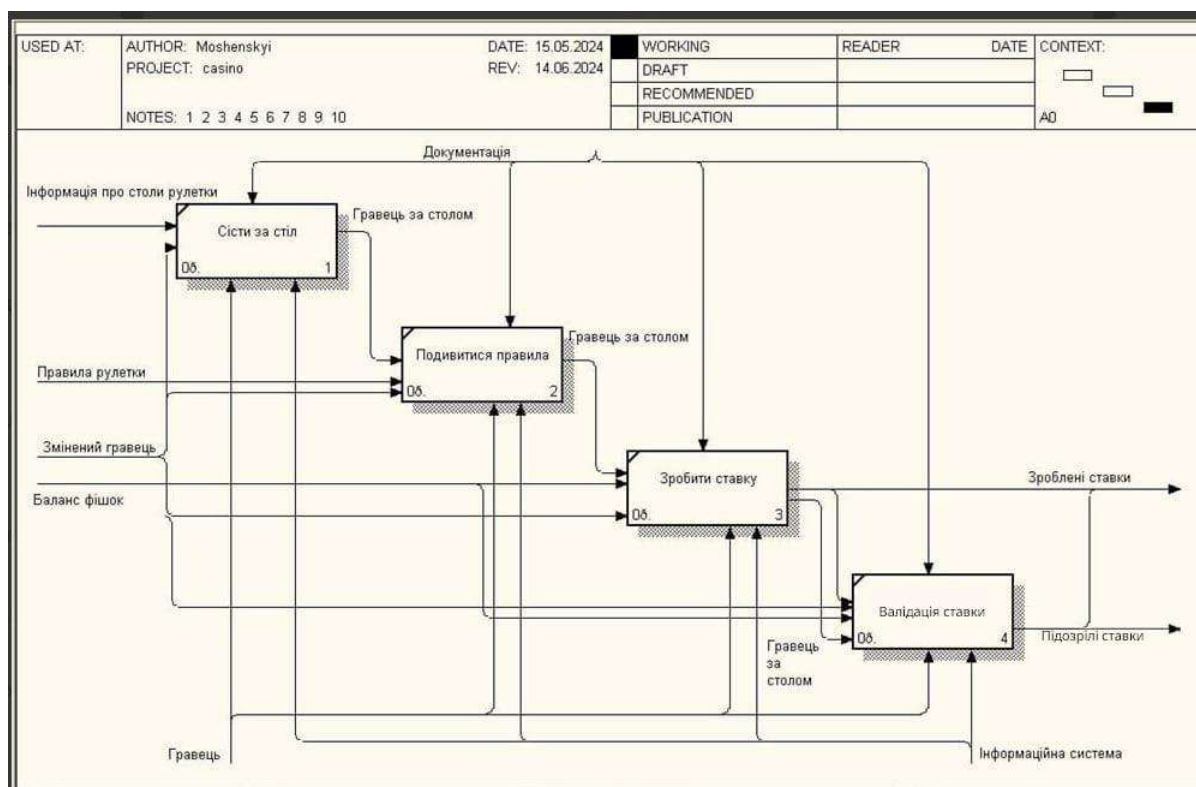


Рисунок 4.4 – IDEF0 Діаграма декомпозиції блоку «Гра в блекджек»

Блок «Сісти за стіл» забезпечує можливість гравця зайняти місце за ігровим столом. Цей блок містить такі дані управління:

«Документація» — нормативні правила та вимоги, що регулюють діяльність ігрового казино на платформі GTA.

Бізнес-функція «Сісти за стіл» має такі дані механізму:

«Гравець» — користувач, який пройшов авторизацію, має створений обліковий запис та достатньо коштів для участі в ігрових процесах;

«Інформаційна система» — система казино, у межах якої здійснюється взаємодія між гравцями для реалізації головної функції.

Бізнес-функція «Сісти за стіл» має такі входні дані:

«Інформація про столи рулетки» — стрілка містить відомості про доступні столи, їхні параметри, розташування та мінімальну ставку.

Бізнес-функція «Сісти за стіл» має такі вихідні дані:

«Гравець за столом» — стрілка відображає, що гравець обрав стіл та отримав можливість розміщувати ставки, скасовувати їх і ознайомлюватися з відповідними правилами.

Блок «Подивитися правила» надає гравцю можливість переглянути правила рулетки, ознайомитися з доступними варіантами ставок і коефіцієнтами виплат. Цей блок містить такі дані управління:

«Документація» — правила та вимоги, що визначають порядок проведення гри в рулетку на платформі GTA.

Бізнес-функція «Подивитися правила» має такі дані механізму:

«Гравець» — авторизований користувач, який має доступ до функціоналу казино;

«Інформаційна система» — програмна система казино, що забезпечує взаємодію гравця з ігровими процесами.

Бізнес-функція «Подивитися правила» має такі вхідні дані:

«Гравець за столом» — стрілка містить інформацію про те, що гравець уже обрав стіл і може переглядати правила;

«Правила рулетки» — стрілка містить повний набір даних про правила гри: можливі ставки, комбінації, коефіцієнти виграшу, часові параметри та інші умови.

Бізнес-функція «Подивитися правила» має такі вихідні дані:

«Гравець за столом» — стрілка передає інформацію про те, що гравець перебуває за вибраним столом і може ознайомлюватися з правилами, розміщувати або скасовувати ставки.

Блок «Зробити ставку» надає гравцеві можливість сформувати ставку відповідно до доступного балансу фішок. Цей блок містить такі дані управління:

«Документація» — нормативні правила та вимоги, що регламентують діяльність ігрового казино на платформі GTA.

Бізнес-функція «Зробити ставку» має такі дані механізму:

«Гравець» — авторизований користувач із власним обліковим записом та достатньою кількістю коштів для участі в грі;

«Інформаційна система» — програмна система казино, у якій організовано процеси взаємодії гравців та реалізації ігрового функціоналу.

Бізнес-функція «Зробити ставку» має такі вхідні дані:

«Гравець за столом» — стрілка, що містить інформацію про те, що гравець обрав стіл та має доступ до виконання ставок;

«Баланс фішок» — стрілка, що відображає актуальний стан кількості фішок, які можуть бути використані для здійснення ставок.

Бізнес-функція «Зробити ставку» має такі вихідні дані:

«Гравець за столом» — стрілка, що підтверджує можливість гравця продовжувати виконувати дії за столом;

«Зроблені ставки» — стрілка, що містить дані про створені гравцем ставки, які змінюються залежно від того, чи додає гравець нову ставку, чи скасовує вже існуючу.

Блок «Забрати ставку» дозволяє гравцю повернути поставлені фішки, якщо вони вже були розміщені на полі. Цей блок містить такі дані управління:

«Документація» — закони, правила та норми, що регулюють діяльність ігрового казино на платформі GTA.

Бізнес-функція «Забрати ставку» має такі дані механізму:

«Гравець» — користувач, який має акаунт, пройшов авторизацію та може виконувати ігрові дії;

«Інформаційна система» — система казино, у межах якої здійснюється обробка ігрових операцій.

Бізнес-функція «Забрати ставку» має такі вхідні дані:

«Гравець за столом» — стрілка, яка засвідчує, що гравець перебуває за вибраним столом і може скасувати власні ставки;

«Баланс фішок» — стрілка, що містить інформацію про кількість фішок після попередніх операцій купівлі або продажу.

Бізнес-функція «Забрати ставку» має такі вихідні дані:

«Гравець за столом» — стрілка, що підтверджує подальшу можливість гравця здійснювати дії за столом;

«Зроблені ставки» — стрілка, що містить оновлену інформацію про ставки після їх часткового або повного скасування гравцем.

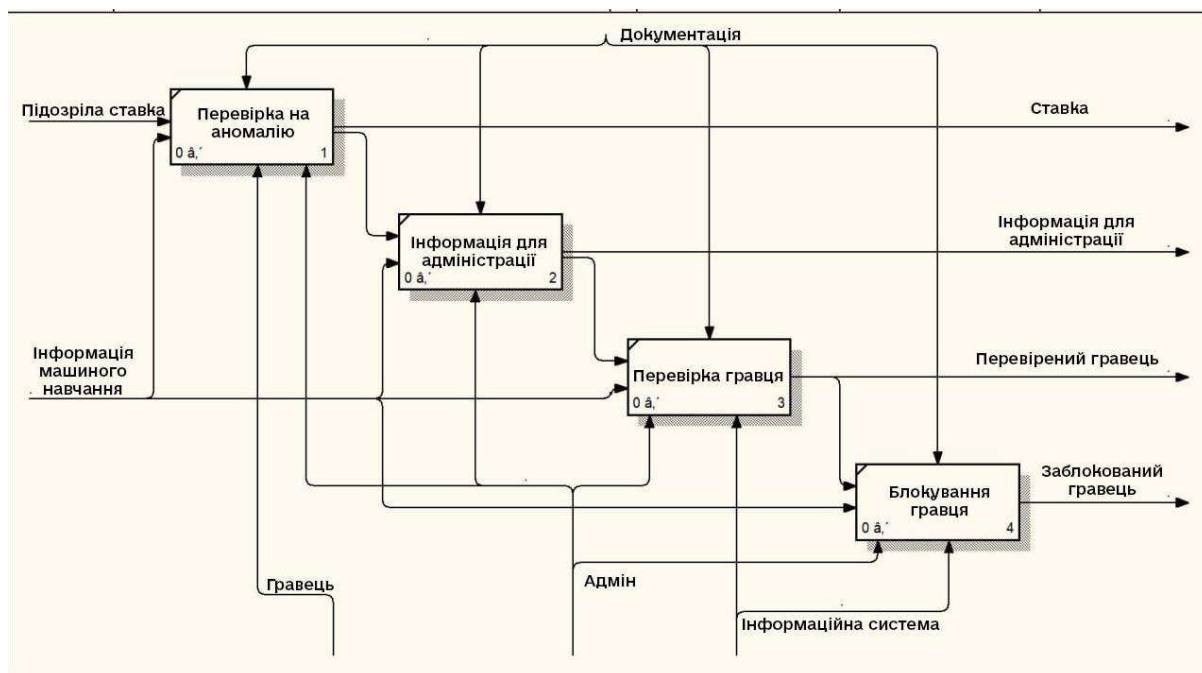


Рисунок 4.5 – IDEF0 Діаграма декомпозиції блоку «Валідація ставки»

Блок «Перевірка на аномалію» відповідає за визначення підозрілих дій гравця та автоматичний аналіз його ставок на основі алгоритмів машинного навчання.

Цей блок містить такі дані управління:

«Документація», тобто закони, правила та регламенти, що визначають критерії виявлення аномалій та регулюють діяльність ігрової системи.

Бізнес-функція «Перевірка на аномалію» має такі дані механізму:

«Гравець» – користувач, який здійснив ставку та взаємодіє з ігровою системою.

«Інформаційна система» – програмний комплекс казино, який відповідає за аналіз ставок та передачу інформації в інші блоки.

Бізнес-функція «Перевірка на аномалію» має такі вхідні дані:

«Підозріла ставка». Ця стрілка містить дані про ставку, що не відповідає типовій поведінці гравця.

«Інформація машинного навчання». Ця стрілка містить дані від ML-моделей, які визначають рівень ризику конкретної ставки.

Бізнес-функція «Перевірка на аномалію» має такі вихідні дані:

«Інформація для адміністрації». Ця стрілка містить результати аналізу та сигналізує про можливі порушення.

«Ставка». Ця стрілка містить інформацію про те, що ставка перевірена та може бути прийнята системою.

Блок «Інформація для адміністрації» забезпечує передачу даних адміністраторам щодо підозрілої активності, що потребує ручної перевірки.

Цей блок містить такі дані управління:

«Документація», тобто регламенти адміністративної роботи та інструкції щодо модерації порушень.

Бізнес-функція «Інформація для адміністрації» має такі дані механізму:

«Адмін» – користувач із правами контролю та реагування на порушення.

«Інформаційна система» – система казино, яка формує та передає службову інформацію адміністрації.

Бізнес-функція «Інформація для адміністрації» має такі вхідні дані:

«Інформація для адміністрації». Ця стрілка містить дані з інших блоків про підозрілу діяльність.

Бізнес-функція «Інформація для адміністрації» має такі вихідні дані:

«Інформація для адміністрації». Ця стрілка містить узагальнену або деталізовану інформацію, яка передається для подальших дій адміністратора.

Блок «Перевірка гравця» виконує оцінку поведінки користувача, визначає можливі порушення та приймає рішення щодо подальшої взаємодії гравця з системою.

Цей блок містить такі дані управління:

«Документація», тобто правила та норми, що регламентують поведінку гравців та визначають критерії підстав для блокування.

Бізнес-функція «Перевірка гравця» має такі дані механізму:

«Гравець» — користувач, що проходить перевірку після зафіксованої аномалії.

«Адмін» — адміністратор, який може підтвердити або скасувати рішення системи.

«Інформаційна система» — система казино, що аналізує історію ставок, статистику та технічні дані гравця.

Бізнес-функція «Перевірка гравця» має такі вхідні дані:

«Інформація для адміністрації». Ця стрілка містить результати попередніх перевірок і повідомлення про підозрілу активність.

«Гравець». Ця стрілка містить інформацію про користувача, якого необхідно додатково перевірити.

Бізнес-функція «Перевірка гравця» має такі вихідні дані:

«Перевірений гравець». Ця стрілка містить дані про статус користувача після перевірки.

Блок «Блокування гравця» забезпечує технічну реалізацію обмеження доступу користувача до системи у випадку підтвердження порушення.

Цей блок містить такі дані управління:

«Документація», тобто правила та регламенти, які визначають підстави та порядок блокування.

Бізнес-функція «Блокування гравця» має такі дані механізму:

«Адмін» – користувач із необхідними правами для підтвердження чи скасування блокування.

«Інформаційна система» – система казино, що реалізує технічну блокування гравця.

Бізнес-функція «Блокування гравця» має такі вхідні дані:

«Перевірений гравець». Ця стрілка містить дані про користувача, щодо якого прийнято рішення про блокування.

«Інформація для адміністрації». Ця стрілка містить підстави для прийняття рішення про блокування.

Бізнес-функція «Блокування гравця» має такі вихідні дані:

«Заблокований гравець». Ця стрілка містить інформацію про те, що користувача заблоковано.

## 5 ОПИС ПРИЙНЯТИХ ПРОЕКТНИХ РІШЕНЬ

### 5.1 Опис архітектури розробленої системи «Реалізація діяльності ігрового казино на платформі GTA»

Розроблення компонентів системи «Реалізація діяльності ігрового казино на платформі GTA» здійснювалося з використанням технологій .NET Core та мови програмування C#, що інтегруються з платформою RAGE MP. Усі модулі системи були створені відповідно до чотирирівневої архітектури, що забезпечує структурованість, розподіл відповідальностей і масштабованість рішення. Архітектура включає такі рівні [11]:

Рівень представлення — реалізований за допомогою Chromium Embedded Framework (CEF), відкритого API, який дозволяє інтегрувати функціональність веб-браузера у застосунки. На основі технологій HTML, CSS і JavaScript забезпечується відображення інтерфейсу та взаємодія користувача з системою.

Клієнтський рівень — фізичний комп'ютер гравця, на якому виконуються операції з обробки та передачі даних. Рівень функціонує на JavaScript у поєднанні з платформою RAGE MP, що забезпечує зв'язок між користувачем і серверною частиною системи.

Серверний рівень — реалізований мовою C# на платформі .NET Core 6.1 разом із серверними модулями RAGE MP. Він відповідає за виконання логіки казино, обчислення, обробку ігрових подій і взаємодію з базою даних.

Рівень бази даних — функціонує на основі СУБД MySQL і забезпечує збереження, оновлення та доступ до інформації, необхідної для роботи системи.

Загальний вигляд реалізованої системи «Реалізація діяльності ігрового казино на платформі GTA» наведено на рис. 5.1.

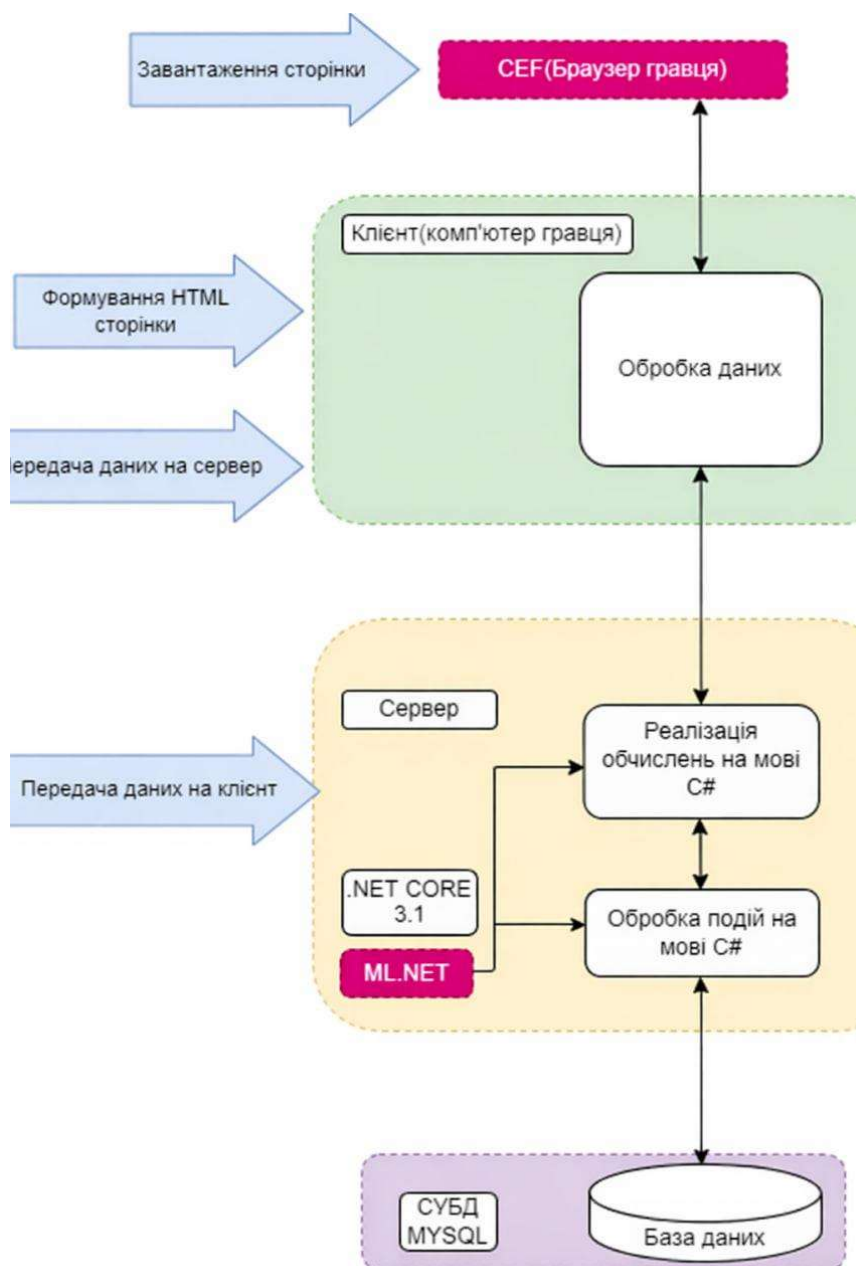


Рисунок 5.1 – Архітектура системи «Реалізація діяльності ігрового казино на платформі GTA»

Опис дій, які відбуваються в системі наступний. Запит від браузера гравця з використання CEF поступає до клієнта на комп'ютері гравця. Клієнт приймає запит від CEF і надсилає його на сервер. Сервер обробляє запит, взаємодіє з базою даних для отримання або збереження даних. База даних MySQL приймає запити від сервера і повертає відповідні дані. Сервер отримує відповідь від бази

даних і формує відповідь для клієнта. Клієнт приймає відповідь від сервера і передає її назад до CEF. Ігровий застосунок у браузері гравця отримує відповідь від клієнта та оновлює інтерфейс користувача. [23]

## 5.2 Обґрунтування вибору мов програмування

Серверна частина використовує мову програмування C# та .NET Core.

На сьогоднішній день C# та .NET Core залишаються одними з найбільш потужних та гнучких технологій для розробки програмного забезпечення у світі. Завдяки широкому набору функцій, ефективній підтримці та активній спільноті розробників, C# та .NET Core є популярними як серед початківців, так і серед професійних розробників у всьому світі.

C# є мовою з суворою типізацією, що дозволяє уникати багатьох помилок під час компіляції, що підвищує надійність коду. Його синтаксис інтуїтивно зрозумілий і добре структурований, що робить його легким для вивчення і використання. Великий набір стандартних бібліотек дозволяє швидко розробляти функціональні додатки.

.NET Core є кросплатформною рамкою, що забезпечує гнучкість при розробці. Він підтримує роботу на різних операційних системах, таких як Windows, Linux, і macOS. Це дозволяє розробникам створювати додатки, які можуть працювати в різних середовищах без необхідності внесення змін до коду.

.NET Core відомий своєю високою продуктивністю та масштабованістю. Це робить його ідеальним вибором для розробки як невеликих веб-додатків, так і великих корпоративних систем. Завдяки своєму модульному дизайну, розробники можуть використовувати лише ті компоненти, які потрібні для конкретного проекту, що сприяє оптимізації продуктивності.

C# та .NET Core мають відкритий вихідний код, що означає, що вони абсолютно безкоштовні для використання. Це значно знижує вартість розробки програмного забезпечення, оскільки немає необхідності витрачати кошти на ліцензії чи дороге програмне забезпечення.

Використання C# і .NET Core надає розробникам великий контроль над своїми проектами. Завдяки потужним інструментам, таким як Visual Studio, та розширеним можливостям відлагодження і тестування, розробники можуть створювати високоякісні додатки з мінімальними зусиллями.

Отже, потужні можливості, кросплатформна підтримка, економічна ефективність та легкий доступ роблять C# та .NET Core ідеальними для розробки сучасних програмних рішень.

Клієнтська частина використовує мову програмування Java Script.

На сьогоднішній день JavaScript залишається однією з найбільш універсальних та практичних мов програмування у світі. Широкі функціональні можливості, дивовижний набір бібліотек та фреймворків, відкритий вихідний код та потужна підтримка онлайн-спільноти зробили JavaScript постійним фаворитом як серед новачків, так і серед авторитетних розробників у світі.

JavaScript вважається однією з найпростіших мов для вивчення. У порівнянні з іншими мовами програмування, JavaScript не вимагає інтенсивного вивчення. Синтаксис JavaScript є логічним і добре організованим. Навіть функції та методи легко зрозуміти, оскільки вони інтуїтивно повідомляють розробнику, яку функцію виконують. В результаті розробникам дуже легко створювати та оптимізувати програми.

JavaScript дуже гнучкий, як під час поточного проекту, так і після його завершення. Гнучкість мови сценаріїв дуже важлива, оскільки функціональність може змінюватися будь-коли під час проекту. Найкраща частина JavaScript – це можливість вносити зміни навіть після запуску проекту, що заощаджує дорогоцінний час. Розробнику не потрібно писати нові коди або функції, оскільки можна вносити зміни до існуючих і використовувати їх.

JavaScript виходить за межі веб-розробки і активно використовується в різних сферах.

JavaScript є мовою з відкритим вихідним кодом, тому він абсолютно безкоштовний. Немає витрат, пов'язаних із придбанням дорогих ліцензій або

програмного забезпечення. Він може ефективно працювати з різними базами даних та серверними технологіями

Порівняно з іншими мовами програмування, JavaScript дозволяє розробнику мати більший контроль. Інші мови програмування можуть бути обмеженими складними синтаксисами, але це не стосується JavaScript. Достатньо кількох простих рядків коду. Крім того, JavaScript дозволяє використовувати різні бібліотеки та фреймворки.

Отже, простота використання, легка інтеграція, економічна ефективність та легкий доступ роблять JavaScript ідеальним вибором для розробки сучасних програмних рішень у різних сферах, включаючи серверні додатки, мобільні додатки, Інтернет речей та настільні додатки.

### 5.3 Обґрунтування вибору СУБД

Система управління базами даних (СУБД) – це програмно-лінгвістичне середовище, що забезпечує створення, зберігання, обробку та управління даними в інформаційних системах. Вона виконує критично важливі функції, серед яких:

MySQL – це система управління реляційними базами даних, яка зберігає дані у вигляді таблиць, забезпечуючи високу структурованість і організованість даних.

MySQL є системою з відкритим вихідним кодом, що дозволяє будь-кому встановити та використовувати базове програмне забезпечення безкоштовно. Це також дозволяє третім сторонам змінювати та налаштовувати вихідний код. Крім того, MySQL забезпечує високу сумісність із широким спектром систем, мов програмування та моделей баз даних. Це робить MySQL простим і практичним варіантом для багатьох організацій.

MySQL розроблена для забезпечення високої продуктивності та швидкості роботи з даними. Це досягається завдяки оптимізації запитів та індексації даних. Вона підходить для різних застосувань, від невеликих додатків до великих веб-сайтів і корпоративних систем.

MySQL забезпечує високу доступність даних завдяки підтримці кластерів і реплікації. Це дозволяє базі даних продовжувати працювати навіть у разі збою окремих серверів, що є важливим для інтернет-компаній та веб-платформ, які повинні надавати цілодобові послуги для глобальної аудиторії.

MySQL може масштабуватися вертикально та горизонтально. Вона підтримує розподіл навантаження та реплікацію даних, що дозволяє обробляти великі обсяги даних і високе навантаження без втрати продуктивності. Це робить MySQL придатною для використання у великих системах і додатках.

MySQL пропонує різноманітні інструменти безпеки, включаючи шифрування даних, контроль доступу, аутентифікацію користувачів та аудит дій. Це дозволяє забезпечити захист конфіденційних даних і захист від кібератак. Також підтримується налаштування політик безпеки для збереження цілісності даних.

В ігровій індустрії MySQL використовується для зберігання та управління різноманітними даними, такими як дані про користувачів, результати ігор, транзакції, інвентар, статистика та багато іншого. Висока продуктивність і швидкість MySQL забезпечують оперативний доступ до даних, що важливо для реального часу обробки запитів у багатокористувацьких іграх.

MySQL використовується для зберігання облікових записів користувачів, профілів, налаштувань та історії ігор. Це дозволяє легко керувати великою кількістю користувачів та їх даними, забезпечуючи швидкий доступ до інформації.

MySQL зберігає різноманітні дані про ігри, включаючи результати, досягнення, інвентар гравців, ігрову статистику та багато іншого. Це дозволяє ефективно обробляти великі обсяги даних і забезпечувати оперативний доступ до них під час гри.

MySQL використовується для збору та аналізу даних про поведінку гравців, їх активність та взаємодію з ігровим контентом. Це дозволяє розробникам ігор отримувати цінні інсайти для покращення ігрового процесу та оптимізації продукту.

MySQL забезпечує безпечне та швидке зберігання даних про внутрішньоігрові покупки, транзакції та платежі. Це важливо для ігор з моделлю монетизації через мікротранзакції, де швидкість і надійність обробки даних мають велике значення.

Завдяки своїй гнучкості, масштабованості та високій продуктивності, MySQL є відмінним вибором для розробки та підтримки ігрових систем, що вимагають надійного управління великими обсягами даних та високої швидкості обробки запитів.

#### 5.4 Логічне та фізичне моделювання даних системи «Реалізація діяльності ігрового казино на платформі GTA»

Під час аналізу розроблюваної системи «Реалізація діяльності ігрового казино на платформі GTA» спроектовано логічну модель даних, що представлена на рис. 5.2.

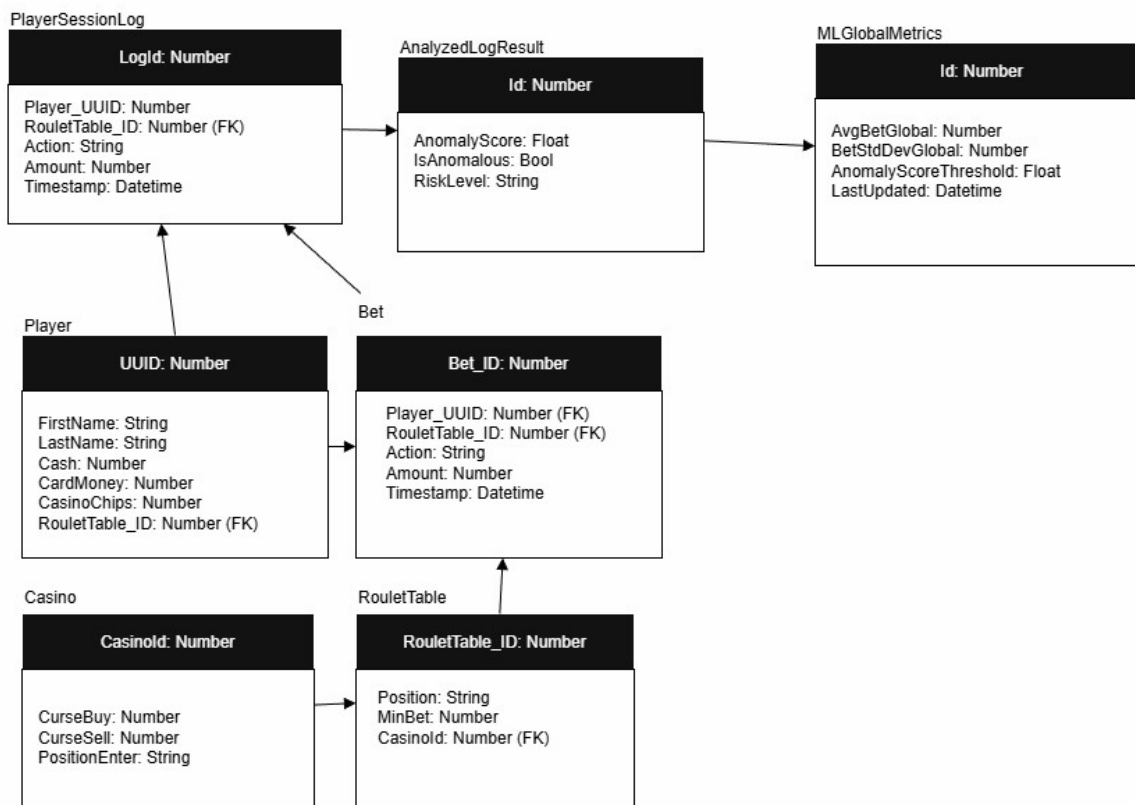


Рисунок 5.2 – Логічна модель системи «Реалізація діяльності ігрового казино на

## платформі GTA»

Логічна модель даних для системи «Реалізація діяльності ігрового казино на платформі GTA» включає визначений набір сутностей з відповідними атрибутами:

сутність Player, що зберігає інформацію про гравця;

сутність Bet, що зберігає інформацію про зроблені гравцем ставки;

сутність PlayerSessionLog, що зберігає історію дій гравця під час сесій;

сутність AnalyzedLogResult, що зберігає результати ML-аналізу логів;

сутність MLGlobalMetrics, що зберігає глобальні метрики для моделі ML;

сутність RouletteTable, що зберігає інформацію про всі ігрові столики;

сутність Casino, що зберігає дані про всі можливі казино.

Типи даних атрибутів, з яких складається сутність Player:

UUID – числовий тип;

FirstName – рядковий тип;

LastName – рядковий тип;

Cash – числовий тип;

CardMoney – числовий тип;

CasinoChips – числовий тип;

RouletteTable\_ID – числовий тип.

Типи даних атрибутів, з яких складається сутність Bet:

Bet\_ID – числовий тип;

Player\_UUID – числовий тип;

RouletteTable\_ID – числовий тип;

Action – рядковий тип;

Amount – числовий тип;

Timestamp – тип дата/час.

Типи даних атрибутів, з яких складається сутність PlayerSessionLog:

LogId – числовий тип;

Player\_UUID – числовий тип;

RouletTable\_ID – числовий тип;

Action – рядковий тип;

Amount – числовий тип;

Timestamp – тип дата/час.

CardsInHand – рядковий тип;

Типи даних атрибутів, з яких складається сутність AnalyzedLogResult:

Id – числовий тип;

AnomalyScore – дробовий тип;

IsAnomalous – логічний тип;

RiskLevel – рядковий тип.

PlayerLogId - числовий тип

Типи даних атрибутів, з яких складається сутність MLGlobalMetrics:

Id – числовий тип;

AvgBetGlobal – числовий тип;

BetStdDevGlobal – числовий тип;

AnomalyScoreThreshold – дробовий тип;

LastUpdated – тип дата/час.

Типи даних атрибутів, з яких складається сутність RouletteTable:

RouletTable\_ID – числовий тип;

Position – рядковий тип;

MinBet – числовий тип;

CasinoId – числовий тип.

Типи даних атрибутів, з яких складається сутність Casino:

CasinoId – числовий тип;

CurseBuy – числовий тип;

CurseSell – числовий тип;

PositionEnter – рядковий тип.

#### 5.4 Створення бази даних системи «Реалізація діяльності ігрового казино на платформі GTA»

Розроблена фізична модель системи «Реалізація діяльності ігрового казино на платформі GTA», зображена на рис. 5.4, має наступні зв'язки:

зв'язок `player-bet` встановлює обмеження на маніпуляції з записами батьківської таблиці `player` у дочірній таблиці `bet` і має тип посилальної цілісності `RESTRICT` на видалення. Це означає, що якщо спробувати видалити запис із таблиці `player` при наявності відповідних записів у таблиці `bet`, буде видана помилка. Також зв'язок `player-bet` має посилальну цілісність типу `CASCADE` на оновлення, яка передбачає автоматичне оновлення пов'язаних записів у таблиці `bet` при зміні даних у таблиці `player`.

зв'язок `roulettetable-bet` встановлює обмеження на маніпуляції з записами батьківської таблиці `roulettetable` у дочірній таблиці `bet` і має тип посилальної цілісності `RESTRICT` на видалення. Це означає, що якщо спробувати видалити запис із таблиці `roulettetable` при наявності відповідних записів у таблиці `bet`, буде видана помилка. Також зв'язок `roulettetable-bet` має посилальну цілісність типу `CASCADE` на оновлення, яка передбачає автоматичне оновлення пов'язаних записів у таблиці `bet` при зміні даних у таблиці `roulettetable`.

зв'язок `roulettetable-player` встановлює обмеження на маніпуляції з записами батьківської таблиці `roulettetable` у дочірній таблиці `player` і має тип посилальної цілісності `RESTRICT` на видалення. Це означає, що якщо спробувати видалити запис із таблиці `roulettetable` при наявності відповідних записів у таблиці `player`, буде видана помилка. Також зв'язок `roulettetable-player` має посилальну цілісність типу `CASCADE` на оновлення, яка передбачає автоматичне оновлення пов'язаних записів у таблиці `player` при зміні даних у таблиці `roulettetable`.

зв'язок `casino-roulettetable` встановлює обмеження на маніпуляції з записами батьківської таблиці `casino` у дочірній таблиці `roulettetable` і має тип посилальної цілісності `RESTRICT` на видалення. Це означає, що якщо

спробувати видалити запис із таблиці casino при наявності відповідних записів у таблиці roulettetable, буде видана помилка. Також зв'язок casino-roulettetable має посилальну цілісність типу CASCADE на оновлення, яка передбачає автоматичне оновлення пов'язаних записів у таблиці roulettetable при зміні даних у таблиці casino.

зв'язок player-playersessionlog встановлює обмеження на маніпуляції з записами батьківської таблиці player у дочірній таблиці playersessionlog і має тип посилальної цілісності RESTRICT на видалення. Це означає, що якщо спробувати видалити запис із таблиці player при наявності відповідних записів у таблиці playersessionlog, буде видана помилка. Також зв'язок player-playersessionlog має посилальну цілісність типу CASCADE на оновлення.

зв'язок roulettetable-playersessionlog встановлює обмеження на маніпуляції з записами батьківської таблиці roulettetable у дочірній таблиці playersessionlog і має тип посилальної цілісності RESTRICT на видалення та CASCADE на оновлення.

зв'язок playersessionlog-analyzedlogresult встановлює обмеження на маніпуляції з записами батьківської таблиці playersessionlog у дочірній таблиці analyzedlogresult і має тип посилальної цілісності RESTRICT на видалення та CASCADE на оновлення.

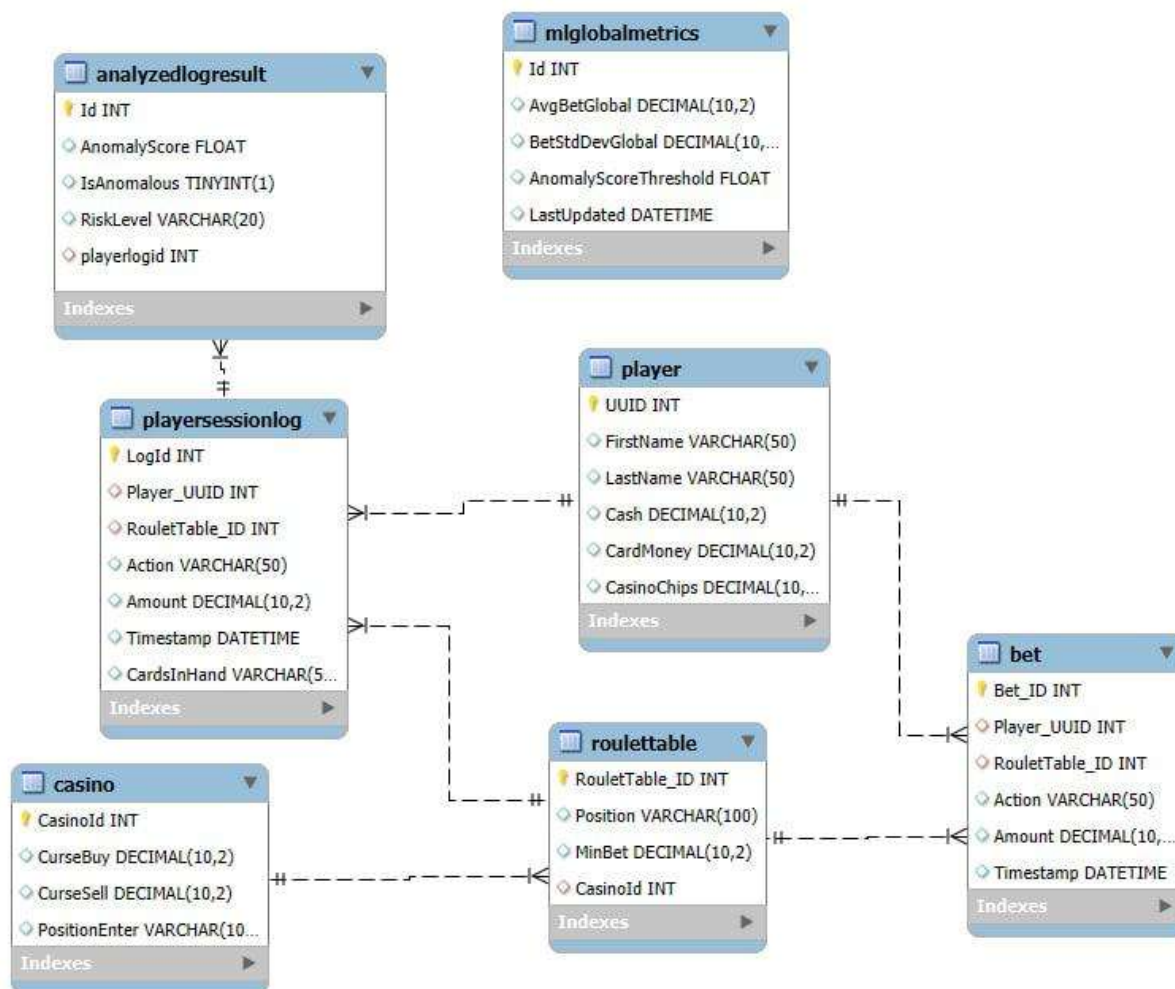


Рисунок 5.4 – Схема даних системи «Реалізація діяльності ігрового казино на платформі GTA»

Таблиця 5.1 – Сутності фізичної моделі даних

№	Назва сутності	Назва атрибуту	Тип даних	Призначення
1	player	UUID	INT	Первинний ключ гравця
		FirstName	VARCHAR(50)	Ім'я гравця
		LastName	VARCHAR(50)	Прізвище гравця
		Cash	DECIMAL(10,2)	Кількість готівки гравця
		CardMoney	DECIMAL(10,2)	Кошти на банківській картці
		CasinoChips	DECIMAL(10,2)	Загальна кількість фішок у гравця
2	casino	RouletteTable_ID	INT (FK)	Зовнішній ключ на roulettetable (поточний стіл)
		CasinoID	INT	Первинний ключ казино
		CurseBuy	DECIMAL(10,2)	Курс купівлі фішок
		CurseSell	DECIMAL(10,2)	Курс продажу фішок
3	roulettetable	PositionEnter	VARCHAR(100)	Координати/точка входу в казино
		RouletteTable ID	INT	Первинний ключ ігрового

				столу
		Position	VARCHAR(100)	Координати столу
		MinBet	DECIMAL(10,2)	Мінімальна допустима ставка
		CasinoID	INT (FK)	Зовнішній ключ на казино
4	bet	Bet ID	INT	Первинний ключ ставки
		Player_UUID	INT (FK)	Зовнішній ключ на player
		RouletTable_ID	INT (FK)	Зовнішній ключ на roulettetable
		Action	VARCHAR(50)	Дія гравця під час ставки (наприклад, «bet placed», «double», тощо)
		Amount	DECIMAL(10,2)	Сума ставки
		Timestamp	DATETIME	Час створення ставки
5	playersessionlog	LogId	INT	Первинний ключ логів дій гравця
		Player_UUID	INT (FK)	Зовнішній ключ на player
		RouletTable_ID	INT (FK)	Зовнішній ключ на roulettetable
		Action	VARCHAR(50)	Дія гравця під час сесії
		Amount	DECIMAL(10,2)	Сума, пов'язана з дією
		Timestamp	DATETIME	Час дії
		CardsInHand	VARCHAR(50)	Поточні карти або комбінація в грі
6	analyzedlogresult	Id	INT	Первинний ключ
		AnomalyScore	FLOAT	Оцінка аномальності (0–1)
		IsAnomalous	TINYINT(1)	Позначка: підозріло чи ні (0 або 1)
		RiskLevel	VARCHAR(20)	Рівень ризику: low, medium, high
		playerlogid	INT (FK)	Зовнішній ключ на playersessionlog
7	mlglobalmetrics	Id	INT	Первинний ключ
		AvgBetGlobal	DECIMAL(10,2)	Середнє значення ставки по казино
		BetStdDevGlobal	DECIMAL(10,2)	Стандартне відхилення ставок
		AnomalyScoreThreshold	FLOAT	Поріг, після якого ставка вважається аномальною
		LastUpdated	DATETIME	Час останнього оновлення метрик

Таблиця 5.2 – Сутності фізичної моделі даних

Назва сутності	Поле FK	Назва сутності 2	Поле PK	ON UPDATE	ON DELETE
bet	Player UUID	player	UUID	CASCADE	CASCADE
bet	RouletTable Id	roulettetable	RouletTable Id	CASCADE	CASCADE
bet	Chips Id	casinochips	Chips Id	CASCADE	RESTRICT
player	RouletTableID	roulettetable	RouletTable Id	CASCADE	SET NULL
roulettetable	CasinoID	casino	CasinoID	CASCADE	RESTRICT

## 5.6 Розробка інтерфейсу клієнтської частини

На даному етапі було реалізовано інтерфейсні елементи інформаційної системи ігрового казино на платформі GTA, що забезпечують взаємодію гравця з ігровим середовищем та модулем гри у Блекджек. Усі екрани розроблені з урахуванням функціональних вимог, зручності користування та інтеграції з серверною логікою казино.

На рисунку 5.6 зображено процес входу гравця до казино. Гравець підходить до входу будівлі та за допомогою клавіші взаємодії ініціює вхід у казино. Даний механізм забезпечує контроль доступу до ігрової зони та слугує початковою точкою взаємодії користувача з інформаційною системою казино.



Рисунок 5.6 – Вхід гравця до казино

Після входу до казино гравець має можливість обміняти ігрову валюту на фішки, які використовуються для участі в азартних іграх. На рисунку 6.10 наведено інтерфейс обміну коштів. Користувач може обрати рахунок, з якого здійснюється обмін (готівка або банківська карта), переглянути поточний баланс

та курс обміну, а також вказати кількість фішок для купівлі або продажу. Після підтвердження операції дані передаються на сервер, де здійснюється перевірка коректності операції та оновлення балансу гравця.

На рисунку 5.8 показано ігрову зону з Блекджек-столами. Гравець може підійти до вільного столу та ініціювати взаємодію з дилером. Візуально відображаються дилер, ігровий стіл та доступні місця для гравців, що створює ефект реалістичної присутності у казино.

Після початку ігрової сесії гравець переходить до етапу розміщення ставок. На рисунку 5.9 представлено інтерфейс вибору ставки. Гравцеві доступний набір фішок різного номіналу, а також відображається загальна сума ставки та поточний баланс. Система очікує підтвердження ставки перед початком роздачі карт.

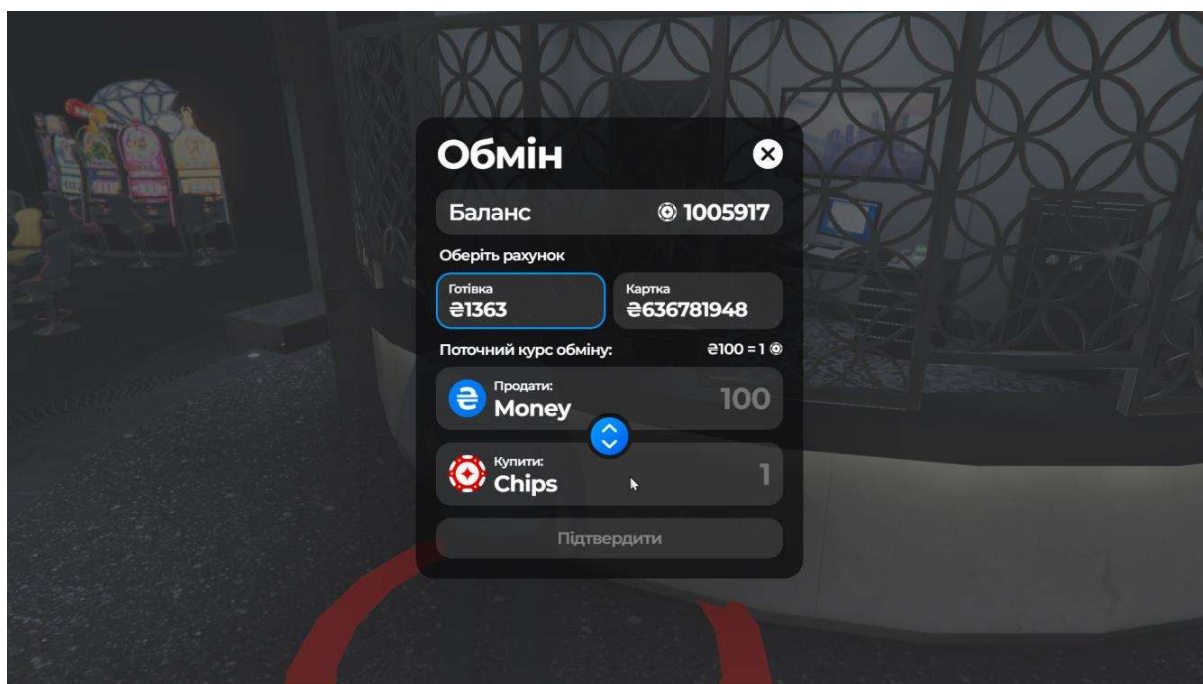


Рисунок 5.7 – Інтерфейс обміну валюти на фішки

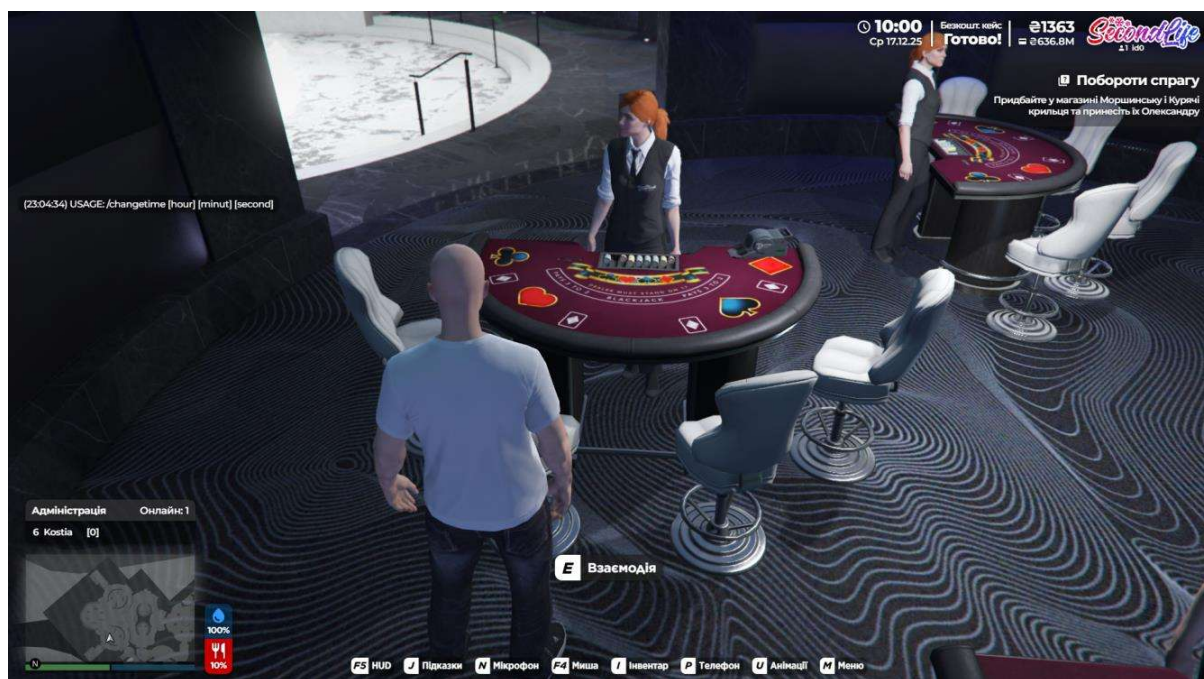


Рисунок 5.8 – Ігрова зона з Блекджек-столами



Рисунок 5.9 – Інтерфейс ставки гравця

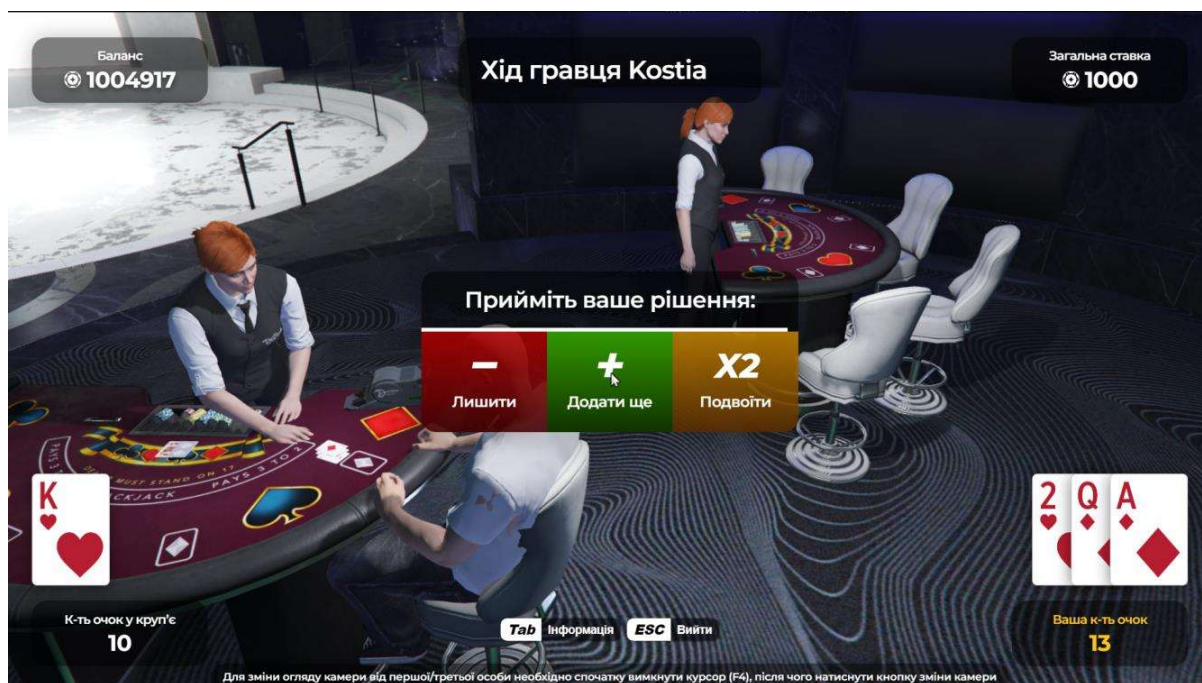


Рисунок 5.10 – Інтерфейс прийняття рішення гравцем у Блекджеку

Після роздачі карт система переходить до етапу прийняття рішення гравцем. На рисунку 6.10 показано інтерфейс вибору дій у Блекджеку. Гравцеві пропонуються стандартні варіанти рішень: взяти ще одну карту, зупинитися або подвоїти ставку. Одночасно відображається поточна кількість очок у гравця та дилера, що дозволяє приймати обґрунтовані рішення під час гри.

Таким чином, реалізований інтерфейс інформаційної системи казино забезпечує повний цикл взаємодії гравця з грою Блекджек: від входу до казино та обміну коштів до розміщення ставок і прийняття ігрових рішень. Інтеграція клієнтської частини з серверною логікою дозволяє коректно обробляти ігрові події, забезпечувати контроль фінансових операцій і формувати основу для подальшого аналізу поведінки гравців.

### 5.7 Тестування розробленого програмного забезпечення

Для перевірки коректності функціонування розробленої інформаційної системи ігрового казино на платформі GTA, а також її відповідності

встановленим функціональним і системним вимогам, було застосовано метод ручного тестування. Сутність даного підходу полягає у виконанні тестових сценаріїв без використання автоматизованих засобів тестування шляхом безпосередньої взаємодії користувача з ігровою системою.

У процесі ручного тестування перевіряючий імітує реальну поведінку гравця, взаємодіючи з ігровими механіками казино, фінансовими операціями та інтерфейсом гри Блекджек з метою виявлення можливих логічних, функціональних або інтерфейсних помилок у роботі програмного забезпечення.

Ручне тестування дозволило перевірити коректність роботи основних модулів системи, зокрема:

- модуля входу до казино та взаємодії з ігровими зонами;
- модуля обміну валюти на ігрові фішки;
- модуля розміщення ставок у грі Блекджек;
- модуля ігрової логіки (роздача карт, підрахунок очок, прийняття рішень гравця);
- модуля обробки фінансових результатів раунду та оновлення балансу гравця;
- модуля фіксації ігрових подій та поведінкової активності користувача.

У межах професійного підходу до тестування було використано заздалегідь підготовлені тестові сценарії, які описують послідовність дій гравця та очікуваний результат виконання кожної операції.

Для проведення тестування обрано такі основні функціональні можливості системи:

- вхід гравця до казино та ініціація взаємодії;
- обмін ігрової валюти на фішки;
- розміщення ставки та участь у грі Блекджек.

Перша функціональна можливість системи – вхід гравця до казино та ініціація взаємодії

Очікувана поведінка:

Після підходу персонажа гравця до входу в казино та натискання клавіші взаємодії система повинна коректно обробити подію та здійснити перенесення гравця до внутрішньої ігрової зони казино. При цьому має відобразитися відповідний інтерфейс та активуватися внутрішні механізми казино.

Передбачаються такі варіанти використання:

– Гравець вперше заходить до казино. Система повинна коректно завантажити інтер'єр, активувати NPC та дозволити доступ до ігрових зон.

– Гравець повторно входить до казино. Система повинна зберігати коректний стан балансу та фішок.

– Гравець намагається взаємодіяти з входом у недоступний час або при помилці сервера. У цьому випадку має відобразитися відповідне повідомлення.

Очікувана швидкодія:

– 1–2 секунди для ініціалізації входу та завантаження казино.

Зауваження:

Функціонал повинен працювати стабільно при повторних входах та не допускати дублювання подій або некоректного стану гравця.

Друга функціональна можливість системи – обмін валюти на ігрові фішки

Очікувана поведінка:

Після взаємодії з касою казино гравцеві має відобразитися інтерфейс обміну, де він може обрати рахунок (готівка або банківська карта), переглянути баланс, поточний курс обміну та вказати кількість фішок для купівлі або продажу. Після підтвердження операції система повинна виконати перевірку достатності коштів та оновити баланс.

Можливі наступні випадки:

– Гравець успішно здійснює обмін. Баланс коштів зменшується, а кількість фішок збільшується.

– Недостатньо коштів для обміну. Система повинна відобразити повідомлення про помилку.

– Гравець скасовує операцію. Баланс залишається без змін.

Очікувана швидкодія:

– 1–2 секунди для виконання операції обміну.

Зауваження:

Фінансові операції повинні виконуватися атомарно, без можливості дублювання або втрати даних.

Третя функціональна можливість системи – розміщення ставки та ігровий процес Блекджек

Очікувана поведінка:

Після підходу до ігрового столу Блекджек та початку сесії гравцеві має бути доступний інтерфейс вибору ставки. Гравець обирає номінал фішок, підтверджує ставку, після чого система виконує роздачу карт та переходить до етапу прийняття рішення.

Можливі наступні випадки:

- Гравець успішно розміщує ставку та бере участь у раунді.
- Гравець не підтверджує ставку вчасно. Раунд пропускається.
- Гравець приймає рішення (взяти карту, зупинитися, подвоїти ставку).

Система коректно обробляє дію та оновлює стан гри.

– Завершення раунду з виграшем, програшем або нічиєю. Баланс фішок оновлюється відповідно до результату.

Очікувана швидкодія:

- до 1 секунди для обробки кожної ігрової дії;
- до 2 секунд для завершення раунду та оновлення балансу.

Зауваження:

Усі ігрові дії повинні реєструватися серверною частиною системи для подальшого аналізу поведінки гравців і контролю коректності роботи казино.

Загальний висновок щодо тестування

У результаті проведення ручного тестування встановлено, що основні функціональні можливості інформаційної системи ігрового казино на платформі GTA працюють коректно та відповідають поставленим вимогам. Система стабільно обробляє взаємодію гравця з казино, фінансові операції та ігрову

логіку Блекджеку, що підтверджує готовність розробленого програмного рішення до подальшої експлуатації та розширення.

### 5.8 Інтеграція ML-модуля в систему

Для реалізації модуля виявлення аномалій у грі блекджек була використана бібліотека ML.NET, яка дозволяє інтегрувати моделі машинного навчання безпосередньо в .NET-екосистему. Основною метою було створити систему, яка може автоматично ідентифікувати нетипові або потенційно шахрайські дії гравців на основі історичних ігрових даних. Враховуючи специфіку задачі, був обраний алгоритм Randomized Principal Component Analysis (PCA), що ефективно виявляє аномалії у багатовимірних числових наборах без потреби в ручному маркуванні.

На етапі підготовки даних для навчання моделі було використано історичні записи, які містили низку структурованих атрибутів: розмір ставки (*Place*), ідентифікатор гравця (*Player\_UUID*), тип ігрового столу (*RouletTable\_Id*), ідентифікатор фішок (*Chips\_Id*), загальну кількість фішок гравця (*CasinoChips*), залишок готівки (*Cash*) та баланс картки (*CardMoney*). На основі цих параметрів формувалася числовий вектор ознак.

Нижче наведено умовні приклади фактичних рядків, які могли бути подані моделі під час тренування:

Таблиця 5.1 – Таблиця даних для навчання

Place	Player UUID	RouletTable Id	Chips Id	CasinoChips	Cash	CardMoney	IsAnomaly
150	321	2	5	4200	3500	12000	0

Пояснення:

Типова гра. Ставка відповідає ресурсам гравця, баланс стабільний. Модель ML.NET визначає таку поведінку як нормальну.

Таблиця 5.2 – Таблиця даних для навчання

Place	Player UUID	RouletTable Id	Chips Id	CasinoChips	Cash	CardMoney	IsAnomaly
4500	321	2	6	4200	3500	12000	1

Пояснення:

Надмірно велика ставка при мінімальному залишку коштів. Цей рядок різко відхиляється від типового розподілу ознак, тому PCA класифікує його як аномальний.

Перед початком тренування була проведена попередня обробка інформації. Зокрема, виконано нормалізацію числових значень для усунення впливу різниці в масштабах (наприклад, між розміром ставки та кількістю фішок). Також здійснено очищення даних: записи з відсутніми або некоректними значеннями видалялися, а числові поля конвертувалися до єдиного формату. Це є критично важливим для стабільної роботи алгоритму PCA, який чутливий до шумів та некоректних значень. Саме навчання проводилося на повністю автоматизованому конвеєрі. Конвеєр складається з етапів перетворення даних та підключеного алгоритму, який намагається зменшити розмірність даних, зберігаючи при цьому їх структуру. Таким чином, модель вчиться розпізнавати основні закономірності та виділяти ті приклади, які значно відхиляються від загального розподілу — тобто аномалії.

Після навчання модель тестувалася на окремій вибірці даних, щоб перевірити її ефективність у виявленні відхилень. Вихідним результатом кожного прогнозу є аналітичний бал аномальності (Anomaly Score), який дозволяє оцінити ймовірність того, що дія гравця є нетиповою. Цей підхід не потребує ручного маркування даних і дозволяє виявляти як навмисні спроби обману системи, так і просто підозрілі або рідкісні стилі гри.

Модель була збережена у вигляді готового файлу, який згодом використовується в основному серверному коді для прогнозування в реальному часі. Завдяки підтримці ML.NET, вся реалізація інтегрується у .NET-додаток без потреби сторонніх інструментів чи мов програмування.

Таким чином, навчання моделі в ML.NET реалізовано як ефективне, масштабоване та цілком кероване рішення, здатне виявляти поведінкові аномалії в грі казино з високим рівнем точності.

## 5.9 Передача даних із MySQL у ML.NET

У реалізованій системі обробки ставок та виявлення аномалій модель ML.NET інтегрована таким чином, щоб мати доступ до актуальних ігрових даних, збережених у базі MySQL. На відміну від потокової чи подієвої моделі, в даному випадку під час запуску серверного застосунку здійснюється початкове зчитування необхідних даних із БД. Ці дані — зокрема інформація про ставки, фішки, гравців та столи — одразу підтягуються у відповідні C#-класи (наприклад, `InputData`) і зберігаються у структурованих колекціях у оперативній пам'яті застосунку.

Завдяки цьому підходу ML-модуль працює швидко, оскільки не виконує запити до БД під час кожного виклику, а звертається до вже підготовлених об'єктів у пам'яті. Клас `InputData` містить набір властивостей, які відповідають ознакам, використаним під час тренування моделі, зокрема: кількість фішок гравця, сума ставки, пов'язаний стіл та інші числові поля. Усі значення приводяться до потрібних типів — наприклад, з типів `INT` або `DECIMAL` у MySQL до `float` чи `double` у .NET.

Якщо під час роботи модулю виникає потреба оновити або оновлювати дані (наприклад, гравець приєднався до нового столу), відповідні колекції можуть бути динамічно оновлені в оперативній пам'яті без повного перезчитування. Таким чином, досягнуто балансу між продуктивністю та актуальністю даних: модель працює з локальними структурами, а не з БД напряму, що значно знижує затримки при прогнозуванні.

Завдяки можливостям ML.NET, прогнозування виконується без потреби формувати проміжні об'єкти `IDataView`. Модель приймає `InputData` безпосередньо, що робить інтеграцію простою та ефективною.

## 5.10 Виявлення аномалій у реальному часі.

ML-модуль було інтегровано в серверне середовище .NET Core таким чином, щоб забезпечити автоматичний виклик моделі при настанні визначених подій і отримання результату в режимі реального часу. При запуску серверного застосунку модель ML.NET завантажується з файла model.zip у пам'ять (за допомогою `MLContext.Model.Load(...)`) і імунізується до використання. Після завантаження формується об'єкт `PredictionEngine<InputData, OutputPrediction>` – «двигун» прогнозування, який інкапсулює модель і дозволяє подавати на вхід одиничні екземпляри даних та отримувати прогноз. Цей `PredictionEngine` реєструється в системі як singleton-сервіс, що буде використовуватися для всіх викликів моделі.

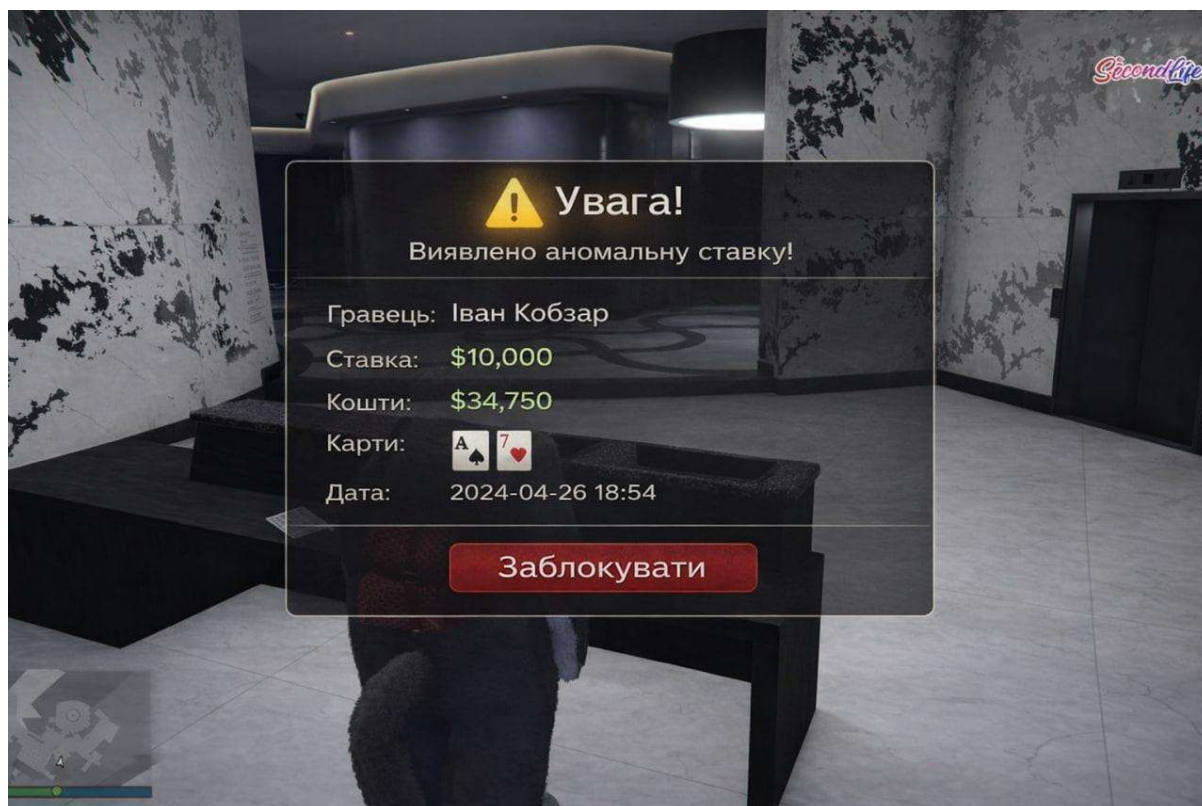
Логіка виклику передбачає, що під час виконання основної ігрової логіки у потрібних місцях відбувається звернення до ML-сервісу. Наприклад, коли гравець робить ставку в казино, серверна функція, що обробляє цю дію, перед завершенням виклику додатково передає дані ставки в ML-модуль. Це реалізовано через метод `mlService.EvaluateBet(inputData)`, який всередині викликає `PredictionEngine.Predict(...)` для отримання результату. Модель повертає результат у вигляді об'єкта `OutputPrediction` – структура, яка містить ключові показники прогнозу. Зокрема, для задачі виявлення аномалій вихід включає булівське поле (`IsAnomaly`) та числовий показник аномального балу (`AnomalyScore`). У системі вихідний клас `PredictionOutput` має поле `IsAnomaly` (позначає, чи віднесена дія до аномальних) та поле `Score` (числовий бал моделі). Після отримання цього результату сервер застосовує бізнес-правила: якщо `IsAnomaly=true` (або ж `Score` перевищує наперед визначений поріг), то подія класифікується як підозріла. Для кожної такої події система виконує журналювання: у базі даних створюється запис про виявлену аномалію – з ідентифікатором гравця, типом дії, значенням ставки, часом та результатами моделі/ Це дозволяє накопичувати історію спрацювань алгоритму та аналізувати їх постфактум.

Важливим аспектом є продуктивність та потоковість цього процесу. Виклик ML-моделі для одиничної дії відбувається дуже швидко (в межах кількох

мілісекунд), тому не створює помітної затримки для гравця. Проте, оскільки сервер казино багатокористувацький, треба врахувати багатопотокове виконання: об'єкт PredictionEngine за замовчуванням не є потокобезпечним. З метою уникнення колізій, у сервісній реалізації використовується один екземпляр PredictionEngine на потік. У підсумку, інтегрований ML-модуль працює у реальному часі паралельно з основною логікою: при кожній новій дії дані негайно аналізуються моделлю, а рішення (аномальна чи ні) приймається та фіксується автоматично.

### 5.11 Сповіщення адміністратора

У впровадженій системі виявлення аномалій ключову роль відіграє своєчасне сповіщення адміністратора про підозрілі дії гравців. На відміну від традиційних рішень із використанням сторонніх служб (електронної пошти, месенджерів або SMS), у даній реалізації система інтегрована безпосередньо у внутрішній інтерфейс гри. Це забезпечує швидке реагування без виходу за межі ігрового середовища.



## Рисунок 5.11 – Інтерфейс адміністратора

Щойно ML-модуль виявляє аномальну ставку — наприклад, нехарактерно високу суму, часту зміну патернів ставок або невідповідну поведінку гравця — адміністрація казино отримує внутрішнє повідомлення в HUD-інтерфейсі гри. Таке повідомлення миттєво з'являється на екрані, і адміністратор може перейти до спеціального меню контролю аномалій.

У цьому меню відображається детальна інформація про інцидент:

1. Нікнейм гравця (який викликав підозру),
2. Сума підозрілої ставки,
3. Поточний баланс гравця (фішки та готівка),
4. Інформація про карти — які саме карти були у гравця на момент аномальної дії (актуально для ігор на зразок блекджека),
5. Дата та час події.

Окрім інформації, меню містить інтерактивну кнопку «Заблокувати», яка дозволяє миттєво заблокувати акаунт або призупинити його дію для подальшого аналізу. Така інтеграція забезпечує не лише оперативність реагування, але й зручність адміністрування: уся необхідна інформація знаходиться в одному місці та доступна в режимі реального часу, без потреби звертатися до сторонніх сервісів чи баз даних.

Цей підхід повністю відповідає вимогам безпеки й стабільності внутрішнього геймплейного середовища, не вимагаючи додаткових залежностей або сервісів. Крім того, його можна масштабувати на інші ігрові сценарії, де важлива оперативна реакція на порушення або спроби шахрайства.

## 6 ПРОВЕДЕННЯ ЕКСПЕРИМЕНТІВ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

### 6.1 Опис проведених експериментів

Для виявлення аномальних ставок у логах казино було проведено серію офлайн-експериментів з використанням двох підходів: моделі машинного навчання (ML.NET) та експертної (правилкової) системи. Перший метод базується на фреймворку ML.NET – фреймворку машинного навчання для C#, і реалізує алгоритм некерованого навчання (unsupervised) для виявлення аномалій. Зокрема, було використано PCA-алгоритм для детекції аномалій (RandomizedPcaTrainer), який будує модель на даних лише одного класу – валідних (нормальних) транзакцій. Під час тренування така ML-модель аналізує статистику типових ставок і формує внутрішню межу «норми». Після навчання для кожної нової ставки обчислюється індекс аномальності (Anomaly Score) та видається бінарний прогноз – аномальна чи ні. Таким чином, модель на основі ML.NET була навчена на історичних нормальних ставках казино, а потім перевірена на тестових даних, що містили суміш нормальних та штучно доданих аномальних ставок. Весь процес здійснювався офлайн, тобто аналізувалися заздалегідь зібрані логи казино (не в реальному часі). Для оцінки якості роботи ML-моделі всі ставки в тестовому наборі було мічено як нормальні або аномальні (штучно, згідно зі сценарієм), що дозволило порівняти передбачення моделі з фактичними мітками і розрахувати показники точності виявлення.

Паралельно було реалізовано експертний (правилковий) підхід – просту систему на основі набору правил для виявлення аномалій. Цей метод не потребує навчання на даних, а визначає аномальні ставки за наперед заданими пороговими умовами. Зокрема, було задано правило, яке помічає ставку як підозрілу, якщо її розмір значно перевищує типовий для даного гравця рівень – наприклад, більше ніж у 10 разів вище за його середню ставку. Друге правило відстежує послідовність дій гравця: якщо гравець здійснив три незвично високі ставки поспіль, така серія розглядається системою як аномальна поведінка. Обрані

порогові значення ( $10\times$  середнього розміру ставки тощо) відповідають явним відхиленням у грі. Експертна правилкова система працює в режимі офлайн на тому ж тестовому наборі даних, що й ML-модель, позначаючи ставки як «аномалія» або «норма» на основі виконання правил. Далі результати двох підходів порівнювалися між собою та з фактичними мітками для оцінки їх ефективності.

## 6.2 Моделювання типових даних ставок та аномалій

Аномальні ставки для тестування моделі та правил генерувалися штучно на основі профілів типового поведінкового патерну гравців. Спершу було змодельовано нормальні ставки: для набору умовних гравців визначено середні очікувані розміри ставок та їх стандартні відхилення. Наприклад, для «обережних» гравців середня ставка становила  $\sim 50\text{--}100$  фішок із невеликим відхиленням (10–20), тоді як для «ризикових» гравців середня ставка могла сягати 200–300 фішок (відхилення 30–50). На основі цих параметрів згенеровано базовий лог нормальних ставок (близько 1000 записів). Далі в цей лог були додані аномальні ставки, які значно перевищують типові для даного гравця значення – як правило, у 2–3 рази вище за норму. Такі аномалії імітують ситуації різкого відхилення від звичного стилю гри. Приміром, гравець, що зазвичай ставить 200–300 фішок, раптом робить ставку у 2000 фішок – це явна аномалія. Серед доданих аномалій були як помірні (близько двократного перевищення середнього рівня), так і екстремальні випадки (в 5–10 разів більші за середнє значення), щоб перевірити чутливість методів до різних ступенів відхилення.

Для тестування ML-моделі аномалій використовувався зазначений набір даних: модель тренувалася лише на нормальних ставках (відповідно до unsupervised-підходу), а потім застосовувалася до тестового набору, що містив ці навмисно додані аномалії. Правилкова система працювала з тим самим тестовим набором: кожна ставка перевірялася на виконання умов правил (наприклад, чи перевищує вона  $10\times$  середнього значення для гравця, чи є вона частиною трьох поспіль великих ставок). Таким чином, обидва підходи аналізували ідентичні

дані, але різними способами: ML-модель обчислювала індекс аномальності для кожної ставки, а експертна система застосовувала фіксовані пороги до значень ставок.

Для наочності на табл. 6.1 наведено приклади кількох ставок із тестового набору та результати їх класифікації ML-моделлю (аномальна чи нормальна). Таблиця містить значення ставок, розрахований для них індекс аномальності (Score) та рішення моделі, порівняне з фактичним статусом ставки; також зазначено, чи було дану ставку правильно класифіковано (True Positive, True Negative) чи допущено помилку (False Positive, False Negative).

Таблиця 6.1 – Сутності фізичної моделі даних

Ставка (фішки)	Індекс аномальності	Моделльний прогноз	Фактична категорія	Результат
300	0.85	Аномалія	Аномалія	TP (вірно)
180	0.60	Аномалія	Аномалія	TP (вірно)
250	0.78	Аномалія	Норма	FP (помилк.)
150	0.40	Норма	Аномалія	FN (помилк.)
100	0.12	Норма	Норма	TN (вірно)

Аналогічно, у табл. 6.2 показано, як експертна правилорова система оцінила ті самі приклади ставок. Тут наведено розмір ставки, рішення правилорової системи (чи визнана ставка аномальною), фактична категорія ставки та результат класифікації (TP, FP, FN, TN).

Таблиця 6.2 – Приклади визначення аномалій експертною (правилоровою) системою

Таблиця 6.1 – Сутності фізичної моделі даних

Ставка (фішки)	Рішення системи	Фактична категорія	Результат
600	Аномалія	Аномалія	TP (вірно)
300	Норма	Аномалія	FN (помилково)
180	Норма	Аномалія	FN (помилково)
250	Норма	Норма	TN (вірно)
150	Норма	Аномалія	FN (помилково)
100	Норма	Норма	TN (вірно)

Як видно з прикладів, правилова система успішно виявляє лише найбільш явні аномалії, проте пропускає менш виражені. Наприклад, ставка 600 фішок (більш ніж у 10 разів перевищує середній рівень обережного гравця) однозначно перевищила поріг і була правильно ідентифікована системою як аномалія (TP). В той же час ставки 300 і 180 фішок, які є аномально високими для своїх гравців (приблизно  $6\times$  та  $3.6\times$  від їх середнього рівня відповідно), не досягли жорсткого порогу  $10\times$  і залишилися невиявленими правилковою системою (FN). Система також не позначила ставку 150 фішок як аномалію (FN), оскільки та лише трохи перевищувала типовий діапазон «обережного» гравця і не відповідала заданим умовам. Водночас нормальна ставка 250 фішок (зроблена гравцем-«ризиком» із високим середнім рівнем) правильно не була позначена експертною системою як підозріла (TN): хоча її абсолютний розмір великий, для цього конкретного гравця вона є звичною, і правило  $10\times$  для нього не спрацювало. З наведених прикладів видно, що використання індивідуального середнього значення гравця в правилах дозволяє уникнути деяких хибних тривог на великі, але нормальні для окремих гравців, ставки (як у випадку 250 фішок, який для ML-моделі був FP[7], а для правилкової системи став TN). Проте жорсткі пороги призводять до того, що значна частина помірних аномалій не виявляється зовсім. Простий набір правил фактично фільтрує тільки найекстремальніші відхилення, тому багато ставок, що виходять за межі норми не радикально, залишаються поза увагою такої системи

### 6.3 Розподіл аномалій за індексом Anomaly Score (ML-модель) та порогами правил

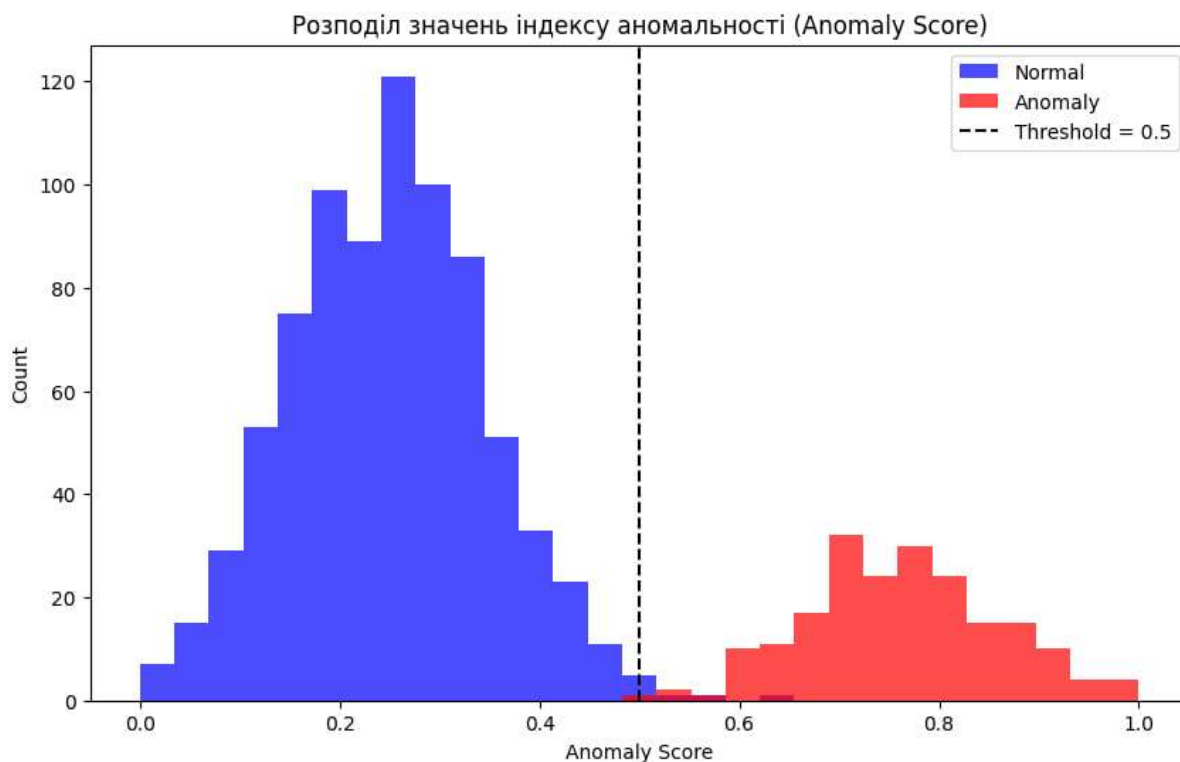


Рис. 6.1 – Гістограму розподілу значень індексу аномальності, отриманих ML-моделлю для тестових даних

На рис. 6.1 показано гістограму розподілу значень індексу аномальності, отриманих ML-моделлю для тестових даних. Гістограма відображає кількість ставок (по вертикалі) в залежності від обчисленого значення Anomaly Score (по горизонталі). Сині стовпчики відповідають нормальним ставкам, а червоні – аномальним (для наочності вибірка аномальних ставок виділена окремо). Як видно з графіку, більшість нормальних ставок мають низький індекс аномальності, зосереджуючись ближче до нуля, тоді як аномальні ставки переважно отримали високі значення Score, зміщені вправо. Між двома цими групами спостерігається певна область перекриття – середні значення Score (близько 0.4–0.6), де можуть траплятися як невеликі аномалії, так і незвично високі, але все ж нормальні для деяких гравців ставки. Для відокремлення

аномалій у моделі було обрано порогове значення індексу (умовно  $\text{Threshold} = 0.5$ ): все, що вище 0.5, модель класифікує як аномалію. На гістограмі цей поріг умовно позначає межу між двома кольорами: видно, що більшість червоних стовпців (аномалій) лежать праворуч від порогу, тоді як сині (норми) – здебільшого ліворуч. Однак у зоні 0.5–0.8 спостерігаються як деякі нормальні ставки з підвищеним Score (це призводить до FP, як у прикладі зі ставкою 250 фішок), так і окремі аномалії з невисоким Score (FN, як ставка 150). Таким чином, розподіл Score підтверджує, що обрана модель загалом правильно розрізняє аномальні дії (вони отримують значно більший індекс), але поріг 0.5 не є ідеальним – частина випадків, близьких до межі, класифікуються з помилками. Надалі цей поріг можна коригувати залежно від пріоритету метрик (знизити для виявлення більшої кількості аномалій або підвищити для зменшення хибних спрацювань).

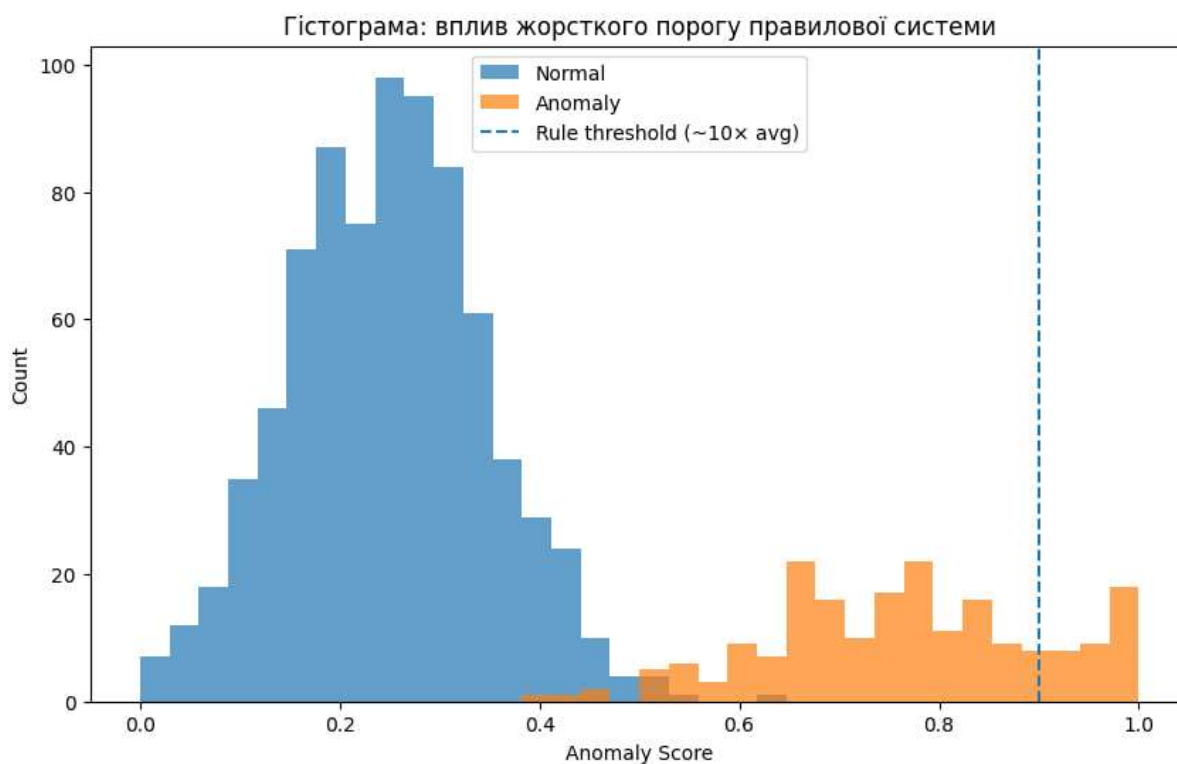


Рис. 6.2 – Гістограму розподілу значень для правилкової системи

Для правилової системи безпосереднього аналога безперервного індексу аномальності немає – кожна ставка або задовольняє умови правила і позначається аномальною, або ні. Втім, результати роботи правил можна зіставити з підходом ML-моделі через еквівалент порогів. Фактично правило «ставка > 10× середнього» аналогічне дуже високому пороговому значенню Anomaly Score, за якого лише найбільш крайні випадки визначаються як аномалії. У нашому експерименті при використанні порогу ~10× середнього для гравця правилова система відфільтрувала майже всі аномальні дії, окрім найбільш екстремальних. В термінах гістограми Score це відповідає ситуації, коли поріг аномальності зміщено далеко вправо: червоні стовпчики (аномалії) в області помірних значень не враховуються, залишаються тільки найбільш праві «хвости» розподілу. При такому жорсткому порозі у правилової системи не виникає зони перекриття – фактично жодна нормальна ставка не досягає настільки великого значення, тож хибних спрацювань нема, але значна частка справжніх аномалій (із нижчим відхиленням) теж не фіксується. Якщо ж знизити пороги в правилах (наприклад, вимагати перевищення середнього не у 10, а у 3–5 разів), то зона перекриття з’явиться: більше аномальних ставок перейде в область детекції, проте деякі нормальні високі ставки або випадкові коливання можуть помилково підпасти під умови правил. Іншими словами, принцип дії правилової системи аналогічний зміні порогу Anomaly Score у моделі: суворіші правила зменшують FP, але дають більше FN, тоді як м’якіші правила підвищують чутливість (Recall), однак можуть збільшувати кількість помилкових тривог.

#### 6.4 Показники точності моделі (Precision, Recall, Accuracy, F1)

Після прогону тестового набору (зі штучно вставленими аномаліями) були розраховані основні метрики якості для побудованої ML-моделі

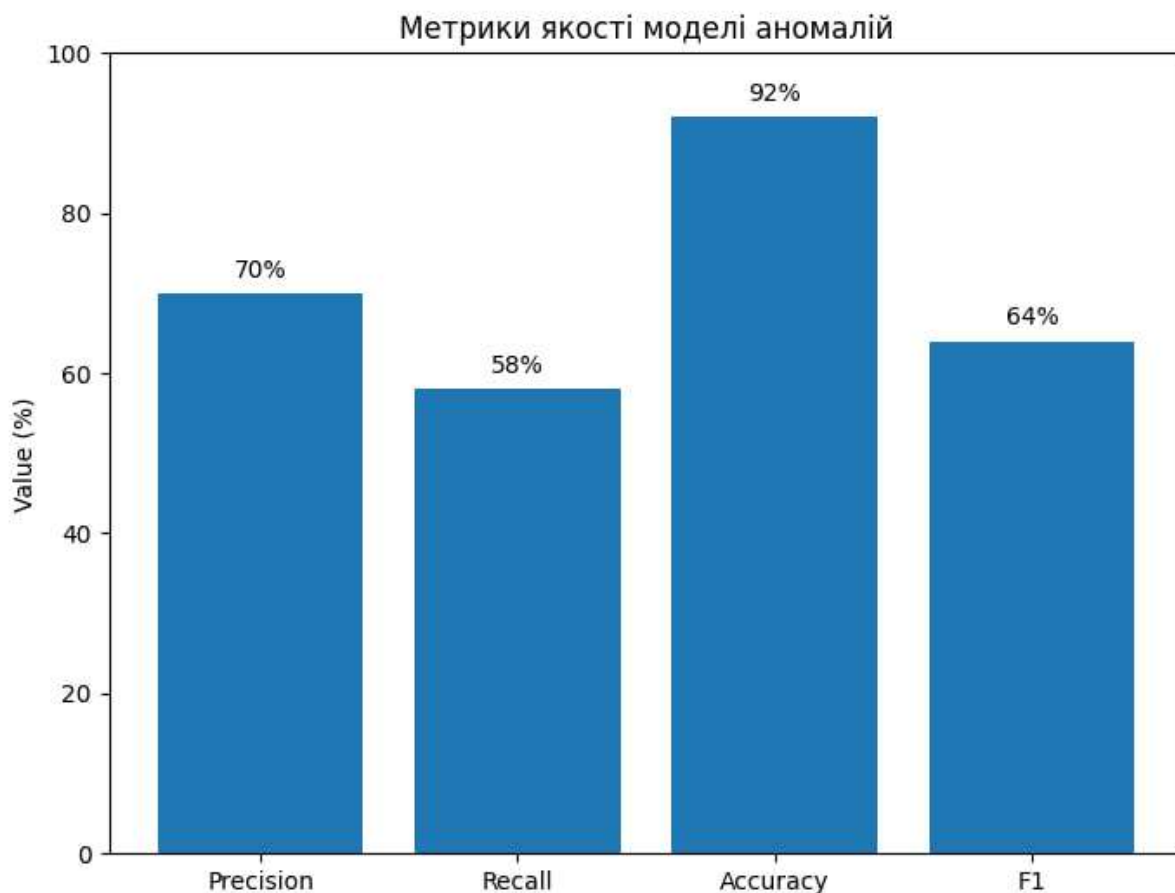


Рис. 6.3 – Метрики якості моделі аномалій: точність (Precision), повнота (Recall), частка правильних класифікацій (Accuracy) та F1-міра.

Як видно з діаграми на рис. 6.3, модель на основі ML.NET продемонструвала такі показники на тестових даних: Precision  $\approx 70\%$ , Recall  $\approx 58\%$ , Accuracy  $\approx 92\%$  та F1  $\approx 64\%$ [16]. Це означає, що із усіх позначених системою підозрілих ставок близько 70% виявилися справді аномальними (точність 70%)[17]. Іншими словами, приблизно третина спрацювань моделі була хибною (False Positive  $\approx 30\%$ ), що вказує на наявність помилкових тривог. З іншого боку, Recall  $\approx 58\%$  свідчить, що система виявила трохи більше половини всіх аномальних дій – решта небезпечних випадків залишилися непоміченими (False Negative  $\approx 42\%$ )[18]. Високе значення Accuracy ( $\approx 92\%$ ) в даному експерименті є очікуваним, адже частка нормальних ставок значно перевищує

кількість аномалій; навіть пропустивши частину аномалій, модель правильно класифікувала більшість прикладів, тому загальна частка правильних класифікацій залишилася високою. Однак, як зазначалося, Accuracy є оманливо високим при незбалансованих даних – успішно розпізнавши 100% нормальних дій, система уже забезпечує великий вклад в Accuracy, навіть якщо знайде лише половину аномалій. Тому ключовими показниками ефективності тут є Precision та Recall. Отримане значення  $F1 \approx 0.64$  (64%) відображає компроміс між точністю та повнотою моделі. По суті, F1-показник середнього рівня означає, що модель має певний баланс: вона достатньо акуратно позначає підозрілі ставки (не надто багато зайвих помилкових тривог), але й пропускає значну частку аномалій, а отже потребує вдосконалення для підвищення чутливості.

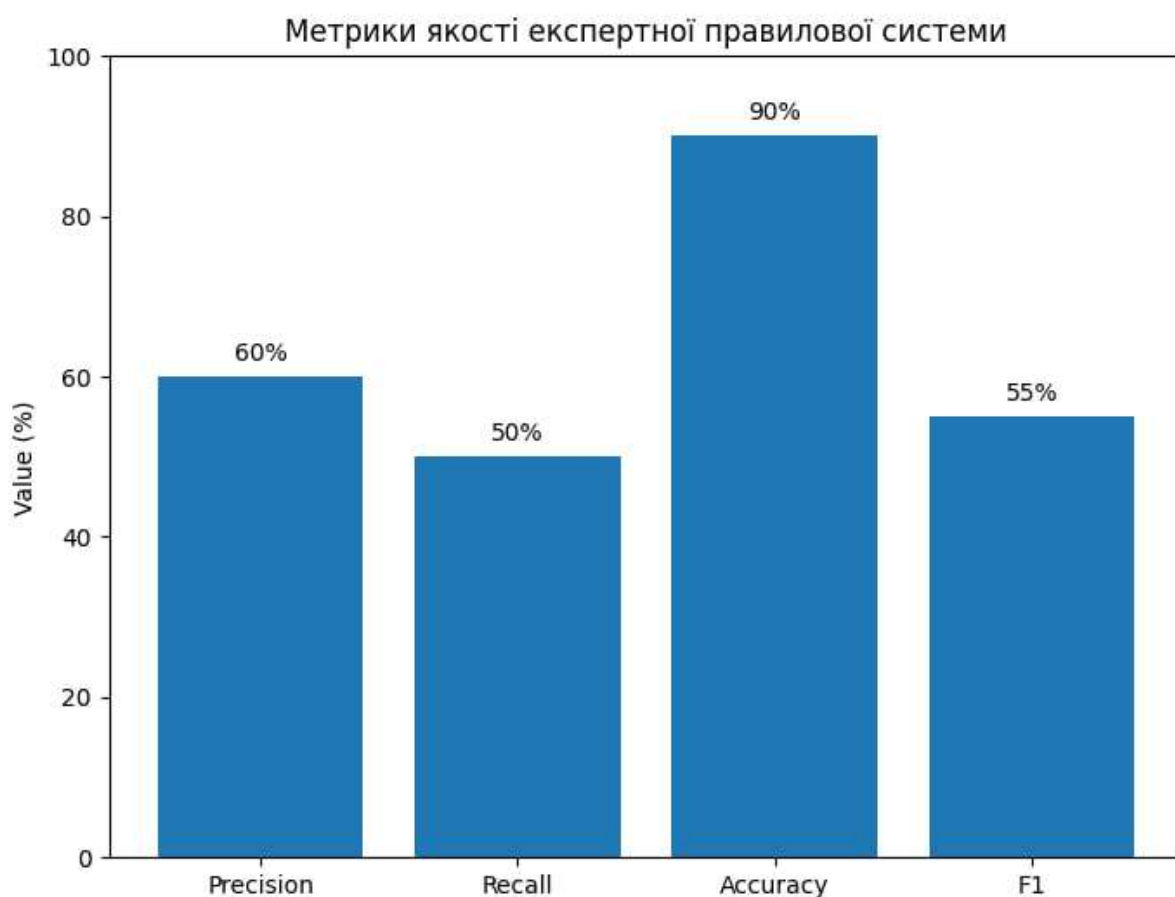


Рис. 6.4 – Метрики якості правової моделі аномалій: точність (Precision), повнота (Recall), частка правильних класифікацій (Accuracy) та F1-міра.

Аналогічні метрики було обчислено і для експертної правилкової системи на тих самих тестових даних. Отримані показники виявилися помітно гіршими, ніж у ML-моделі: Precision  $\approx 60\%$ , Recall  $\approx 50\%$ , Accuracy  $\approx 90\%$ , F1  $\approx 55\%$ . Це означає, що із усіх ставок, позначених правилковою системою як аномальні, лише  $\sim 60\%$  насправді були аномаліями (точність  $\sim 60\%$ ), тобто частка хибних спрацювань склала близько 40%. Одночасно система виявила лише приблизно половину реальних аномальних дій (Recall  $\sim 50\%$ ), пропустивши решту підозрілих випадків. Сукупна точність класифікації (Accuracy  $\approx 90\%$ ) залишилася високою головним чином через велику частку правильних класифікацій нормальних ставок (правилова система майже не чіпала більшість звичайних дій, особливо при високих порогах). Низьке значення F1  $\sim 55\%$  підтверджує значно гірший баланс між precision та recall для правилкового підходу.

Відзначимо, що наведені метрики для експертної системи залежать від налаштування порогів правил. Вказані вище значення ( $\sim 60/50/90/55$ ) відповідають відносно «м'якому» налаштуванню, при якому правила трохи знижені для виявлення більшої кількості аномалій (що дало певну кількість FP). Якби використовувалося максимально консервативне правило (наприклад, поріг  $10\times$  без додаткових умов), Precision міг би досягати  $\sim 100\%$  за рахунок повної відсутності хибних тривог – але Recall був би вкрай низьким (менше 30%). Таким чином, правила доводиться підлаштовувати, шукаючи компроміс, але навіть при оптимізації балансу їх показники залишаються гіршими, ніж у ML-моделі.

## 6.5 Інтерпретація результатів та приклади помилок

Результати експериментів показують, що обидва підходи здатні виявляти аномальні дії гравців, проте кожен має свої сильні та слабкі сторони. ML-модель (ML.NET) продемонструвала збалансованішу ефективність: її Precision  $\sim 70\%$  свідчить про відносно невелику кількість хибних спрацювань – система рідко позначає нормальні ставки як підозрілі. Це важливо для практичного використання, адже мінімізує зайві перевірки чесних гравців. З іншого боку,

Recall  $\sim 58\%$  означає, що помітна частка аномалій лишається невиявленою. Пропущені аномалії (False Negative) потенційно становлять найбільшу загрозу, оскільки небезпечна поведінка може залишитися без уваги. В нашому випадку модель не помітила 5 з 12 аномальних ставок у тесті (наприклад, ставка 150 фішок була FN). Ці пропущені випадки зазвичай стосуються менш виражених відхилень – коли розмір ставки лише трохи перевищує типову варіацію гравця. Якщо поріг аномальності встановлено досить високим (щоб уникати FP), модель може не реагувати на такі помірні відхилення.

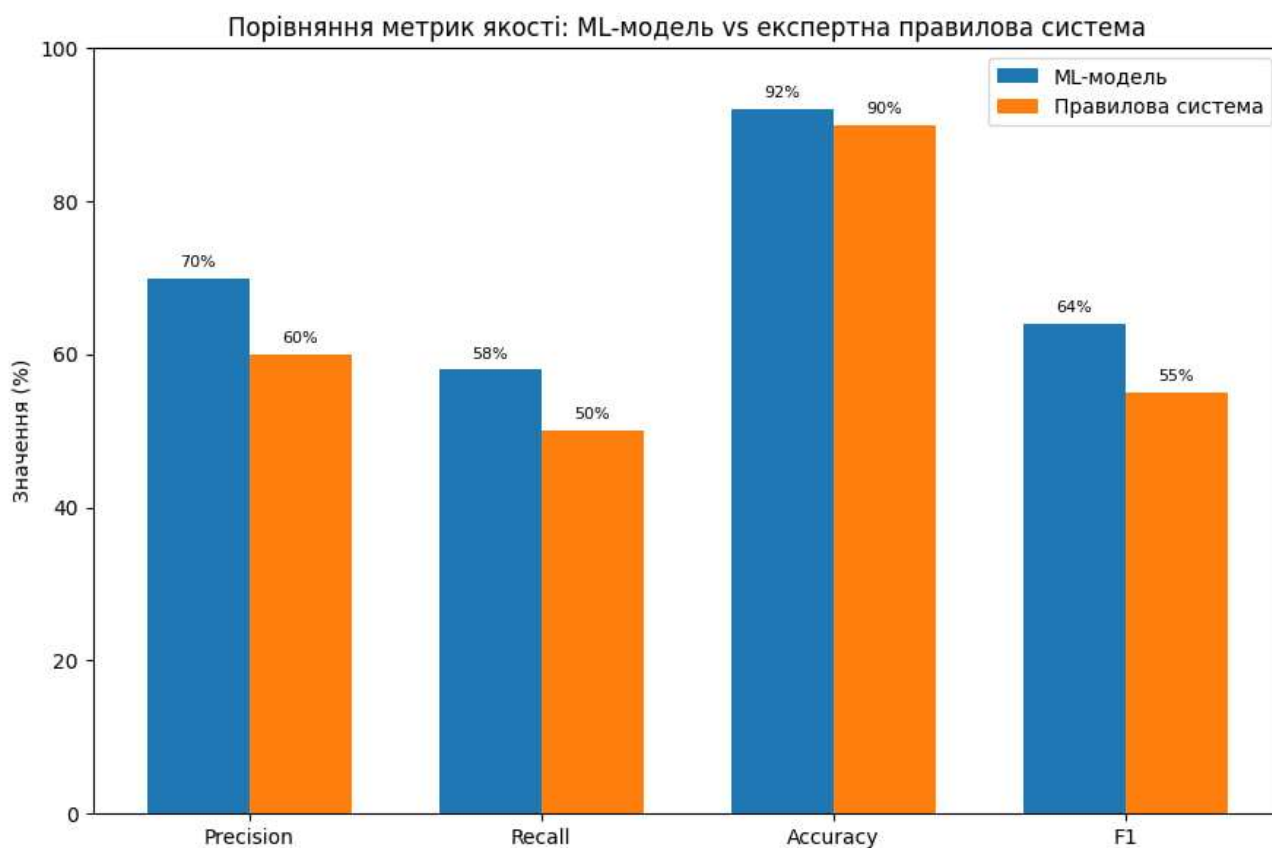


Рис. 6.5 – Порівняння метрик якості

Експертна (правилова) система, своєю чергою, проявила інший баланс характеристик. Перевагою правилкового підходу є його простота та «обережність» щодо хибних тривог: при достатньо суворих порогах (наприклад, правило  $10\times$ ) система практично не позначає нормальні дії як аномальні. У наших експериментах правилкова система дійсно рідко помилково сигналізувала про аномалію на звичайних ставках, особливо коли використовувався високий поріг – це позитив, адже мінімізує зайві перевірки. Наприклад, ставка 250 фішок від гравця з високим середнім рівнем не була позначена правилами як підозріла, і жодної хибної тривоги на подібні великі, але допустимі для цього гравця ставки система не згенерувала (на відміну від ML-моделі, для якої ця ставка стала FP)

Однак ціна такої високої специфічності – дуже низька чутливість: правилкова система пропустила багато аномальних дій, тобто має низький Recall. Ми бачили, що при правилі  $10\times$  більшість штучно доданих аномалій не досягли порогу і залишились невиявленими (FN). Навіть при помірнішому налаштуванні (як у наших розрахунках метрик, де Recall  $\sim 50\%$ ) експертний підхід не зміг знайти майже половину аномалій у вибірці. Пропущені випадки – це саме ті помірні відхилення, які не вписалися в жорсткі рамки правил. Наприклад, ставки 150–300 фішок для обережних гравців виявились занадто великими для моделі (їх Score був підвищений, і модель деякі з них виявила), проте для правил  $10\times$  вони не виглядали достатньо екстремальними, тож система їх проігнорувала (FN). Це підтверджує, що фіксований глобальний множник (типу  $10\times$ ) не охоплює всі можливі аномальні відхилення. Якщо ж знизити пороги, щоб виявляти такі менш виражені аномалії, виникає протилежна проблема – з'являються хибнопозитивні спрацювання.

У наших досліджах помірно послаблення правил (порог  $\sim 5\times$  або додаткові умови на зразок 3 великих ставок поспіль) дозволило підвищити Recall правилкової системи до  $\sim 50\%$ , але при цьому точність впала до  $\sim 60\%$ . Це означає, що близько 40% спрацювань правил в такому режимі були хибними. Такі False Positive у експертній системі зазвичай виникають через природні коливання у поведінці гравців, які прості правила помилково інтерпретують як аномалію.

Наприклад, якщо гравець зіграє три відносно високі для себе ставки підряд просто в рамках свого стилю або стратегії, правило «3 великі ставки поспіль» може спрацювати, хоча насправді гравець не вчиняв нічого шахрайського. Таким чином, правилочий підхід стикається з труднощами у налаштуванні балансу між чутливістю і специфічністю: жорсткі правила дають мало хибних тривог, але багато пропусків, а гнучкіші правила виявляють більше аномалій, зате генерують зайві тривоги.

Порівнюючи два підходи безпосередньо, можна відзначити, що ML-модель загалом спрацювала краще на нашому наборі даних. Вона виявила більшу частку аномалій, ніж експертна система, при відносно близькому рівні хибних спрацювань. Зокрема, ML-модель забезпечила Recall  $\sim 58\%$  проти  $\sim 50\%$  у правил, тобто знайшла більше підозрілих дій. Precision моделі також вищий ( $\approx 70\%$  проти  $60\%$ ), що вказує на меншу частку помилкових тривог. Як наслідок, F1-міра для ML-підходу істотно перевищує відповідний показник правилочої системи ( $\sim 64\%$  проти  $\sim 55\%$ ), що означає кращу узгоджену якість виявлення.

Втім, в окремих ситуаціях прості правила можуть спрацювати краще. Уже зазначалося, що використання індивідуального середнього показника в правилах захистило експертну систему від деяких FP, які допускала глобальна ML-модель. Тобто правилочий підхід враховує контекст конкретного гравця простим способом – через його середнє значення ставки. ML-модель у поточній реалізації такого контексту явно не мала (вона тренувалася на загальному масиві без атрибутів гравця), через що виникла проблема: одна глобальна межа аномалії не змогла охопити різнотипних гравців. Для високоваріативних «ризикових» гравців модель генерувала FP (помилково сигналізуєчи їх великі, але нормальні ставки як аномалії), а для надто стабільних гравців – FN (не помічаючи помірні аномалії на тлі малого розкиду ставок). Експертна система частково пом'якшує цю проблему, застосовуючи пороги відносно середнього кожного гравця. У результаті для гравців з високими лімітами вона не реагує на їх ставки типового для них рівня (уникаючи деяких FP), і водночас для низьколімітних гравців будь-яке різке перевищення їх невеликої бази легко ловиться правилом. Таким чином,

у аспекті врахування індивідуальних відмінностей прості правила в чомусь перевершили базову ML-модель.

Крім того, правилний підхід може бути налаштований на виявлення конкретних шаблонів, наприклад «серії агресивних ставок», які поточна ML-модель (що аналізує кожну ставку окремо) може й не розпізнати. Якщо б шахрайська поведінка проявлялася не в одиничній ставці, а в певній послідовності дій, добре продумане правило здатне це підхопити. У наших даних основна увага зосереджувалась на розмірах ставок, тож ML-модель і правила орієнтувалися на величини; але в принципі правила можна розширити на часові або послідовні патерни (хоча це ускладнить систему і виходить за межі базового набору).

Незважаючи на окремі успіхи, обмеження експертної системи є суттєвими. Вона покриває лише ті випадки, які явно передбачені умовами правил. Якщо з'явиться новий тип підозрілої поведінки, неописаний чинними правилами, така система його просто не помітить. Розробнику доведеться вручну додавати нові правила або коригувати існуючі пороги. З ростом кількості правил система може стати надто складною, правила можуть конфліктувати чи вимагати пріоритетизації. Більше того, фіксовані порогові значення не враховують всіх нюансів розподілу даних – як ми бачили, один глобальний множник не підходить одночасно для всіх гравців, а підбирати індивідуальні параметри для десятків і сотень гравців вручну практично неможливо.

ML-підхід в цьому плані є набагато гнучкішим і перспективнішим для розширення. По-перше, модель машинного навчання можна удосконалити, забезпечивши їй додаткові поведінкові ознаки гравця (середній розмір ставки, дисперсію, категорію гри тощо), або ж виконати сегментацію гравців і тренувати окремі моделі для різних груп. Такі підходи вже були згадані у рекомендаціях (розд. 6.6) і дозволять моделі враховувати контекст автоматично, зменшуючи кількість FP без втрати Recall. По-друге, ML-модель можна тренувати на дедалі більшому обсязі реальних даних казино, включаючи різноманітні сценарії, що підвищить її здатність вловлювати нетипову поведінку. На відміну від статичних

правил, алгоритм навчання здатен підлаштовуватися під нові види загроз: достатньо доповнити вибірку прикладами нової шахрайської схеми, і модель потенційно навчиться її розпізнавати. По-третє, підтримка та розширення ML-системи зводиться до періодичного перенавчання або оновлення моделі, тоді як експертна система вимагала б постійного ручного супроводу правил. У підсумку, хоча прості правила можуть слугувати непоганим базовим рішенням чи швидким фільтром для найбільш грубих порушень, саме підхід на основі машинного навчання є більш перспективним для подальшого розвитку. Він забезпечує кращу загальну ефективність уже на базовому рівні і має великий потенціал для підвищення точності за рахунок глибшого аналізу даних, гнучкого налаштування під різні умови та масштабування під реальні потоки даних казино.

## ВИСНОВОК

У ході виконання магістерської кваліфікаційної роботи розглянуто та проаналізовано предметну область розробки інформаційних систем для реалізації діяльності ігрових казино на платформі Grand Theft Auto (GTA). Досліджено особливості функціонування віртуальних казино, сучасний стан розвитку ігрових сервісів та підходи до побудови інформаційних систем, що забезпечують інтерактивну взаємодію гравців з ігровим середовищем.

У процесі виконання роботи досягнуто поставленої мети, а саме – розроблено інформаційну систему ігрового казино на платформі GTA, яка забезпечує автоматизацію основних бізнес-процесів казино, зокрема управління ігровими сесіями, фінансовими операціями, взаємодією гравців з ігровими столами та обробкою ігрових подій. Для досягнення цієї мети було проведено аналіз предметної області, досліджено особливості ігрової механіки казино та визначено вимоги до функціонування системи.

У роботі розглянуто архітектурні підходи до побудови ігрових інформаційних систем. На основі проведеного аналізу було обґрунтовано використання багаторівневої архітектури, що складається з клієнтської частини, прикладної логіки та серверного рівня. Клієнтська частина забезпечує взаємодію гравця з ігровим інтерфейсом, прикладний рівень реалізує бізнес-логіку казино та правила ігор, а серверна частина відповідає за зберігання, обробку та захист даних.

У межах кваліфікаційної роботи сформульовано функціональні та нефункціональні вимоги до інформаційної системи казино. Для формалізації процесів функціонування системи побудовано функціональні моделі IDEF0 та діаграми потоків даних (DFD), які дозволили описати ключові процеси взаємодії гравця з казино, обміну коштів, розміщення ставок і проведення ігрових раундів. Також виконано проектування логічної та фізичної моделей бази даних, що забезпечують зберігання інформації про гравців, ставки, ігрові столи, фішки та результати ігрових сесій.

Розроблена інформаційна система дозволяє автоматизувати значну частину рутинних операцій казино, зменшити ймовірність помилок у фінансових розрахунках і підвищити рівень контролю за ігровими процесами. Система забезпечує безпечну обробку транзакцій, коректне оновлення балансу гравців та фіксацію ігрових подій для подальшого аналізу.

Для перевірки працездатності розробленої системи та відповідності її функціонування встановленим вимогам було проведено ручне тестування основних функціональних можливостей, зокрема входу до казино, обміну валюти на фішки, розміщення ставок і реалізації ігрової логіки Блекджеку. Результати тестування підтвердили коректність роботи системи, стабільність її функціонування та придатність до практичного використання у віртуальному ігровому середовищі.

Таким чином, результати виконаної магістерської роботи створюють теоретичну та практичну основу для подальшого розвитку інформаційних систем ігрових казино на платформі GTA, розширення ігрового функціоналу та впровадження інтелектуальних методів аналізу поведінки гравців.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- 1 Мошенський К.О. Розробка та впровадження інформаційної системи ігрового казино в сервер гри GTA. ХНУРЕ, 2024.
- 2 Russell S., Norvig P. Artificial Intelligence: A Modern Approach. 4th ed. Pearson, 2020.  
URL: <https://aima.cs.berkeley.edu/> (дата звернення: 20.12.2025).
- 3 Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2021.  
URL: <https://www.deeplearningbook.org/> (дата звернення: 20.12.2025).
- 4 Chollet F. Deep Learning with Python. 2nd ed. Manning Publications, 2021.  
URL: <https://www.manning.com/books/deep-learning-with-python-second-edition> (дата звернення: 20.12.2025).
- 5 Zhang Y., Li H., Wang J. Fraud Detection in Online Gambling Using Machine Learning. IEEE Access, 2022.  
URL: <https://ieeexplore.ieee.org/> (дата звернення: 20.12.2025).
- 6 OpenAI. GPT-4 Technical Report. 2023.  
URL: <https://arxiv.org/abs/2303.08774> (дата звернення: 20.12.2025).
- 7 Microsoft Azure Machine Learning Documentation. Microsoft Corp., 2025.  
URL: <https://learn.microsoft.com/azure/machine-learning/> (дата звернення: 20.12.2025).
- 8 Rockstar Games. Grand Theft Auto V Server API Documentation. 2021.  
URL: <https://support.rockstargames.com/> (дата звернення: 20.12.2025).
- 9 Kaggle. Blackjack Player Behavior Dataset. 2023.  
URL: <https://www.kaggle.com/> (дата звернення: 20.12.2025).
- 10 ISO/IEC 25010:2023. Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation.
- 11 PostgreSQL Global Development Group. PostgreSQL Documentation. Version 15. 2023.  
URL: <https://www.postgresql.org/docs/15/> (дата звернення: 20.12.2025).

- 12 Flask Documentation. Version 2.3. Pallets Projects, 2024.  
URL: <https://flask.palletsprojects.com/en/2.3.x/> (дата звернення: 20.12.2025).
- 13 React.js Documentation. Meta Platforms Inc., 2024.  
URL: <https://react.dev/> (дата звернення: 20.12.2025).
- 14 OWASP Foundation. Web Security Testing Guide. 2022.  
URL: <https://owasp.org/www-project-web-security-testing-guide/> (дата звернення: 20.12.2025).
- 15 NVIDIA Corporation. GPU Acceleration for Machine Learning. White Paper, 2021.  
URL: <https://www.nvidia.com/en-us/> (дата звернення: 20.12.2025).
- 16 Gartner Inc. Trends in Online Gaming Analytics. Industry Report, 2023.  
URL: <https://www.gartner.com/> (дата звернення: 20.12.2025).
- 17 ACM Digital Library. Adaptive Game Systems and Player Modeling. 2022.  
URL: <https://dl.acm.org/> (дата звернення: 20.12.2025).
- 18 IEEE Spectrum. AI in Gaming: 2025 Outlook. IEEE, 2025.  
URL: <https://spectrum.ieee.org/> (дата звернення: 20.12.2025).
- 19 Харківський національний університет радіоелектроніки. Методичні рекомендації щодо написання та оформлення магістерських кваліфікаційних робіт. ХНУРЕ, 2023.
- 20 КПІ ім. Ігоря Сікорського. Приклади оформлення магістерських робіт. Електронна бібліотека, 2024.  
URL: <https://ela.kpi.ua/> (дата звернення: 20.12.2025).
- 21 Supabase Documentation. Supabase Inc., 2023.  
URL: <https://supabase.com/docs> (дата звернення: 20.12.2025).
- 22 MongoDB Atlas Documentation. MongoDB Inc., 2022.  
URL: <https://www.mongodb.com/docs/atlas/> (дата звернення: 20.12.2025).
- 23 Unity Technologies. Game Server Architecture and Multiplayer Systems. 2021.  
URL: <https://unity.com/> (дата звернення: 20.12.2025).
- 24 OWASP Foundation. Cheat Detection in Online Games. 2023.  
URL: <https://owasp.org/> (дата звернення: 20.12.2025).

25 Google Cloud. BigQuery for Gaming Analytics. Documentation, 2024.  
URL: <https://cloud.google.com/bigquery/docs> (дата звернення: 20.12.2025).