

Міністерство освіти і науки України  
Кам'янець-Подільський національний університет імені Івана Огієнка  
Фізико-математичний факультет  
Кафедра комп'ютерних наук

**Кваліфікаційна робота бакалавра**

з теми:

«Розробка інформаційно-аналітичної системи для опрацювання та візуалізації  
результатів моніторингу успішності студентів»

Виконав: здобувач вищої освіти  
групи KN1-B20  
спеціальності 122 Комп'ютерні науки  
Михайло КОСІНОВ

Керівник:

Олена СМАЛЬКО, кандидат педагогічних  
наук, доцент кафедри комп'ютерних наук

Рецензент:

Олексій ЗЕЛЕНСЬКИЙ,  
кандидат фізико-математичних наук,  
доцент кафедри математики, доцент

Кам'янець-Подільський – 2024 р.

## ЗМІСТ

АНОТАЦІЯ.....	3
ВСТУП.....	5
РОЗДІЛ 1. ОСОБЛИВОСТІ МОНІТОРИНГУ ОСВІТНІХ ДОСЯГНЕНЬ ЗДОБУВАЧІВ ВИЩОЇ ОСВІТИ.....	8
1.1 Необхідність і особливості проведення моніторингу у закладах вищої освіти.....	8
1.2 Критерії та показники для проведення моніторингу успішності.....	14
1.3 Навчальна аналітика в умовах цифровізації освіти.....	16
РОЗДІЛ 2. ВИБІР ТЕХНОЛОГІЙ І ВИКЛИКИ ТЕХНІЧНОЇ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ.....	18
2.1 Існуючі програмні засоби візуалізації статистичних даних.....	18
2.2 Особливості технічної реалізації.....	26
2.3 Інструменти побудови вебзастосунків.....	27
РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ЗАСТОСУНКУ.....	36
3.1 Формування вимог до інформаційно-аналітичної системи.....	36
3.2 Практична реалізація системи.....	39
3.3 Тестування застосунку.....	47
ВИСНОВКИ.....	54
ДОДАТКИ.....	57
Додаток А Зразок джерела даних.....	57
Додаток Б Лістинг обчислення показників.....	58
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	60

## АНОТАЦІЯ

Косінов М. С. Розробка інформаційно-аналітичної системи для опрацювання та візуалізації результатів моніторингу успішності студентів. – Кваліфікаційна (бакалаврська) робота на здобуття освітньо-кваліфікаційного рівня вищої освіти спеціальності 122 “Комп’ютерні науки” освітньо-професійної програми “Комп’ютерні науки та інформаційні технології”. – Кам’янець-Подільський університет імені Івана Огієнка, 2024.

Розглянуто теоретичні основи дослідження щодо необхідності створення інформаційно-аналітичної системи для опрацювання і візуалізації результатів успішності студентів, розкрито необхідність і особливості здійснення локального моніторингу закладу вищої освіти, визначено критерії та показники які необхідно реалізувати у застосунку, розкрито важливість розвитку навчальної аналітики у закладах вищої освіти.

Досліджено існуючі інструменти візуалізації статистичних даних, інструменти створення вебдодатків, розкрито особливості функціональної реалізації інформаційно-аналітичної системи.

Розкрито етапи розробки, описано функціональні можливості та особливості проведення тестування інформаційно-аналітичної системи для опрацювання та візуалізації результатів моніторингу успішності студентів.

*Ключові слова:* “якість вищої освіти”, “академічна аналітика”, “локальний моніторинг”, “моніторинг освітніх досягнень”, “моніторинг успішності”, “інформаційно-аналітична система”, “візуалізація результатів успішності”, “панелі візуалізації”.

## ANNOTATION

Kosinov M. S. Development of an information-analytical system for processing and visualizing student performance monitoring results. – Qualification (bachelor's) paperwork for obtaining the educational and qualification level of higher education in the specialty 122 “Computer Science” of the educational-professional program “Computer Science and Information Technologies”. – Kamianets-Podilskyi Ivan Ohienko University, 2024.

The theoretical foundations of the study on the necessity of creating an information-analytical system for processing and visualizing student performance results are considered. The necessity and specifics of conducting local monitoring in higher education institutions are disclosed, criteria and indicators to be implemented in the application are defined, and the importance of developing educational analytics in higher education institutions is highlighted.

Existing tools for visualizing statistical data, tools for creating web applications, and the specifics of the functional implementation of the information-analytical system are researched.

The stages of development are disclosed, functional capabilities are described, and the features of testing the information-analytical system for processing and visualizing student performance monitoring results are detailed.

Keywords: “quality of higher education”, “academic analytics”, “local monitoring”, “educational achievements monitoring”, “performance monitoring”, “information-analytical system”, “performance results visualization”, “visualization dashboards”.

## ВСТУП

**Актуальність.** Технологічний розвиток суттєво змінив сталі погляди і підходи до виконання рутинних завдань. Сьогодні значна кількість виробничих процесів автоматизовані у різних галузях промисловості, наприклад харчовій, текстильній, машинобудівній, освітній, медичній та інших. Необхідність здійснення ефективного управління та використання даних систем вимагає кваліфікованих і освічених працівників, які здатні приймати швидкі та важливі рішення у короткі терміни створює додаткові вимоги рівня і якості знань, які надають заклади вищої освіти.

Навчання, опосередковане технологіями, набуло помітного значення у вищій освіті, а навчальна аналітика стала потужним інструментом для відстеження взаємодії учнів у системі закладів вищої освіти з метою покращення результатів навчання студентів. Використання інструментів навчальної аналітики з використанням дашбордів (інформаційних панелей) набуває все більшого поширення у закладах освіти у прогресуючих країнах, таких як Сполучені Штати Америки, Німеччина, Іспанія, Велика Британія та інших [1].

У галузях освіти і науки України навчальна аналітика є перспективним напрямом досліджень, результати яких дозволять забезпечити гнучкість навчального процесу на всіх рівнях організації навчального закладу: від керівника установи до кожного окремого студенту. Проєктування систем призначених для навчальної аналітики вимагають комплексного рішення, яке включає в себе компоненти для збору, обробки, аналізу та відображення даних у вигляді звіту, що містить необхідні результати, для проведення детального аналізу на предмет дотримання академічної доброчесності в межах вивчення навчальної дисципліни, правильного використання методів і методик викладання навчального матеріалу. Опрацювання результатів моніторингу навчання дозволяє визначати проблемні і малоефективні ділянки освітнього процесу, надає основу для прийняття управлінських рішень, необхідних для підвищення якості і ефективності навчання.

**Мета роботи** – створити інформаційно-аналітичну системи для опрацювання та візуалізації результатів моніторингу успішності студентів. Для досягнення мети було визначено наступні **завдання**:

1. дослідити особливості моніторингу освітніх досягнень здобувачів вищої освіти;
2. дослідити методики опрацювання результатів моніторингу освітніх досягнень студентів;
3. вивчити існуючі способи та програмні засоби навчальної статистики призначених для опрацювання і візуалізації даних;
4. проаналізувати інструменти побудови вебзастосунків, призначених для опрацювання і візуалізації даних;
5. розробити інформаційно-аналітичну систему для опрацювання та візуалізації результатів моніторингу успішності студентів.

**Об’єкт дослідження** – моніторинг успішності у закладах вищої освіти з використанням інформаційних технологій

**Предмет дослідження** – інструменти побудови вебзастосунків, призначених для опрацювання і візуалізації даних.

**Методи дослідження:**

1. Аналізу і синтезу: виявлення закономірностей у результатах проведеного моніторингу успішності здобувачів вищої освіти, виявлення критеріїв і показників, що дозволяють визначити компоненти для проведення подальшого аналізу.
2. Спостереження: проведення тестування і перевірки працездатності системи з різними наборами тестових даних.
3. Порівняльний: проведення порівняння між існуючими засобами візуалізації статистичних даних з метою виявлення їх переваг і недоліків, щоб використати здобуті відомості під час проєктування інформаційно-аналітичної системи.

4. Моделювання: реалізація прототипу застосунку, необхідного для спостереження за точністю обчислень, визначення практичних результатів.
5. Експерименту: відтворення взаємодії користувача з інформаційно-аналітичною системою для виявлення помилок.

**Практичне значення одержаних результатів:** створена інформаційно-аналітична система візуалізує результати успішності здобувачів освіти у вищих навчальних закладах для проведення подальшого аналізу з метою вдосконалення та виявлення проблем освітнього процесу.

**Апробація одержаних результатів:**

1. Виступ на науковій конференції студентів і магістрантів Кам'янець-Подільського національного університету імені Івана Огієнка за підсумками науково-дослідної роботи у 2023 році.
2. Виступ на II Міжнародній науково-практичній інтернет-конференції «Актуальні аспекти розвитку STEAM-освіти в умовах євроінтеграції» (м. Кропивницький, 26 квітня 2024 року).

**Публікації:**

1. Оpubлікована стаття у віснику Кам'янець-Подільського національного університету імені Івана Огієнка. Фізико-математичні науки [2].
2. Оpubліковані тези у збірнику матеріалів наукової конференції за підсумками науково-дослідної роботи здобувачів вищої освіти фізико-математичного факультету Кам'янець-Подільського національного університету імені Івана Огієнка у 2023-2024 н.р. [3]
3. Публікація тез у збірнику матеріалів II Міжнародної науково-практичної інтернет-конференції (м. Кропивницький, 26 квітня 2024 року) [4].

Дипломна робота включає: анотацію, вступ, три розділи, дев'ять пунктів, висновки, список використаних джерел та додатки. Також у роботі наведено 20 ілюстрацій та продемонстровано лістинги програмного коду.

## **РОЗДІЛ 1. ОСОБЛИВОСТІ МОНІТОРИНГУ ОСВІТНІХ ДОСЯГНЕНЬ ЗДОБУВАЧІВ ВИЩОЇ ОСВІТИ**

Проблема забезпечення якості освіти є актуальним предметом дослідження для науковців у багатьох країнах світу. Це зумовлено постійним еволюційним і технологічним прогресом. Зміни у природних процесах, розвиток технологій та бажання людини осягнути невідомі горизонти Всесвіту покладають велику відповідальність на інтелектуальну еліту суспільства, яка продукує нові знання та винаходи. Україна, зі свого боку відома всьому світу не лише як держава з багатими природними ресурсами, а й потужним інтелектуальним потенціалом. Прикладом цьому слугують видатні науковці і винахідники, такі як І. Сікорський, Б. Патон, В. Вернадський.

Створення нових пристроїв, систем і об'єктів різної природи походження (медичні пристрої, механізми та ін.) вносять зміни до різних галузей діяльності людини і вимагають від неї отримання нових специфічних знань та навичок. На державному рівні, ефективне використання існуючих і здобуття нових знань відіграють важливі роль у зміцненні позицій країни на міжнародній арені. Саме тому, постійність прогресу і рівень якості системи освіти є невід'ємними складовими розвитку держави і поліпшення рівня життя її громадян.

### **1.1 Необхідність і особливості проведення моніторингу у закладах вищої освіти**

Ефективність закладів вищої освіти розглядається як ключовий фактор розвитку сучасної держави. Університети та інститути мають складну організаційну структуру, основними підрозділами якої є факультети, кафедри та відділи забезпечення навчального процесу (бібліотека, центр інформаційних технологій, бухгалтерія та інші). Відповідно до визначення міжнародної організації стандартизації якість – сукупність характеристик об'єкта, що відносяться до його здатності задовольняти встановлені та передбачувані потреби [5]. Зважаючи на те, що якість визначається як сукупність характеристик

певного об'єкту, то для забезпечення високої якості освіти необхідний складний, комплексний і системний процес, який чітко визначає мету, цілі і завдання, що покладаються на заклади освіти усіх рівнів.

Міністерством освіти і науки України опубліковано результати міжнародного дослідження, згідно результатів якого, станом на 2024-й рік, в умовах війни з Росією система освіти України зазнає не лише матеріальних але й освітніх втрат, які визначаються загальним зниженням якості освіти [6]. Аналізуючи результати даного дослідження, допустимо зробити висновки, що проблема якості освіти в Україні не мала такого важливого значення як сьогодні.

Найбільша відповідальність покладена на заклади вищої освіти, оскільки відповідно до Закону України «Про вищу освіту» метою вищої освіти є отримання здобувачем освіти високого рівня наукових, професійних і загальних компетентностей, які необхідні для використання у певному напрямку діяльності [7]. Процес реформування закладів вищої освіти, який триває впродовж останніх років вимагає створення стандартизованої, загальноприйнятої і чітко визначеної системи для здійснення моніторингу і визначення якості освіти відповідно до потреб суспільства у спеціалістах високої кваліфікації для різних сфер виробництва і здійснення соціальної діяльності [8].

На якість освіти впливає значна кількість факторів, серед яких:

- рівень кваліфікації науково-педагогічного складу;
- матеріально-технічне забезпечення навчального закладу;
- наявність необхідної інфраструктури та інструментів необхідних для здійснення науково-дослідницької діяльності;
- наявність тісних комунікативних зв'язків між науково-педагогічним складом і здобувачами освіти;
- рівень залученості здобувачів освіти у життя навчального закладу.

З метою отримання відомостей про стан освітнього процесу і його регулювання у сучасних навчальних закладах здійснюється моніторинг.

Зважаючи на те, що моніторинг є широким поняттям і може використовуватись у різних галузях науки було введено поняття моніторингу якості освіти.

За визначенням Державної служби якості освіти України моніторинг якості освіти – це система послідовних і систематичних заходів, що здійснюються з метою виявлення та відстеження тенденцій у розвитку якості освіти в країні, на окремих територіях, у закладах освіти (інших суб'єктах освітньої діяльності), встановлення відповідності фактичних результатів освітньої діяльності її заявленим цілям, а також оцінювання ступеня, напряму і причин відхилень від цілей [9]. Багатогранність і варіативність системи освіти України вимагає проведення комплексного моніторингу освітніх процесів для забезпечення відповідного рівня якості вищих навчальних закладів.

Здійснення ефективного моніторингу якості вищої освіти вимагає реалізації повноцінного комплексу процесів, які покладаються на науково-педагогічний склад. В першу чергу, для проведення моніторингу необхідне виконання навчально-методичних умов, які включаються в себе [10]:

- реалізація партнерських зв'язків між викладачем і студентом. Основою для цього є спільне виконання наукового, професійного, творчого і громадської видів діяльності;
- структуризація навчального процесу на додаткові складові (педагогічна, соціальна, психологічна та інші);
- інтеграція інформаційних технологій.

Наявність зворотного зв'язку, спільна участь у творчих і професійних заходах студентів та викладачів має вплив на результати проведення моніторингу, що в кінцевому результаті надає можливість коригувати якість освіти. Після побудови тісної соціальної взаємодії і структуризації освітнього процесу відповідно до потреб викладачів і здобувачів освіти стає необхідною інтеграція інформаційних технологій у процес моніторингу. Використання автоматизованих систем для здійснення та візуального представлення результатів систематичного моніторингу успішності за допомогою використання web-технологій надає

додаткову мотивацію суб'єктам освітнього процесу.

Інтеграція систем комплексного моніторингу з метою вдосконалення навчальної діяльності закладів вищої освіти дозволяє отримати широкий перелік переваг, серед яких можна визначити: зростання конкуренції і мотивації до професійного зростання між здобувачами освіти, здатність викладачами швидко вносити зміни у навчальний процес, незалежно від форми навчання (очна, дистанційна) здатність здійснення навчання з мінімізацією негативних наслідків на якість освіти та інші.

Враховуючи виклики, які необхідно подолати для забезпечення високої якості освітніх процесів необхідним було дослідження особливостей проведення і здійснення моніторингу, його етапи і чинники, які є важливою складовою для досягнення мети кваліфікаційної роботи.

За призначенням моніторинг у широкому розумінні поділяється на дві масштабні категорії [11]:

1. Інформаційний. Призначення даного моніторингу полягає у систематизації, зборі і накопиченні інформації стосовно об'єкту спостереження.
2. Управлінський. Кінцевим результатом управлінського моніторингу вважається достатня кількісна і якісна інформація, яка може слугувати для прийняття управлінських рішень та змін у процесах діяльності установ

Для прийняття ефективних і важливих управлінських рішень, які позитивно впливатимуть на розвиток якості освіти в Україні необхідно побудувати чітко визначену, змістовну і неперервну стратегію здійснення моніторингу суб'єктів навчальної діяльності.

Залежно масштабності і цілей здійснення моніторингу, він може здійснюватись на наступних рівнях:

- Державний - передбачає відповідність до єдиних державних стандартів, що накладаються на конкретні навчальні заклади.

- Регіональний – проведення дослідження освітнього об'єкта на рівні області, міста/району.

- Локальний - здійснюється безпосередньо у вищому навчальному закладі.

Проведення локального моніторингу визначається складним і послідовним процесом, який може будуватись за допомогою різних форми і підходів. За формою проведення моніторинг поділяється на наступні категорії:

- самооцінювання науково-педагогічного складу, здобувачів вищої освіти;
- керівниками, в межах провадження діяльності в межах вищого навчального закладу;
- зовнішнє оцінювання органами проведення і здійснення моніторингу освітнього процесу у межах вищого навчального закладу.

Високий професіоналізм викладачів, дотримання правил академічної доброчесності та високий рівень соціальної відповідальності є необхідними, оскільки значущим компонентом якісного та ефективного моніторингу є отримання достовірної та правдивої інформації про стан освітнього процесу. Спираючись на дослідження фахівців, які аналізували особливості проведення моніторингу якості вищої освіти необхідно передбачити виконання наступних етапів [8-9]:

1. Призначення, мета і завдання згідно яких здійснюється моніторинг.
2. Визначити об'єкт за яким здійснюється спостереження.
3. Визначення критеріїв і показників відповідно до яких оцінюється об'єкт моніторингу.
4. Збір і обробка інформації, необхідних для здійснення моніторингу.
5. Опрацювання і обробка даних, отриманих після отримання необхідної інформації.
6. Аналіз результатів моніторингу і прийняття рішень коригування освітнього процесу.

В свою чергу, моніторинг якості освіти можна розподілити на декілька

основних напрямів:

- умов функціонування освітньої системи;
- освітнього процесу;
- моніторинг результатів освітнього процесу.

Допустимо виділити моніторинг результатів освітнього процесу у вигляді фундаментального напрямку, який є однією зі складових інших напрямів. Результати успішності навчання здобувачів вищої освіти тісно пов'язані з іншими освітніми компонентами: науково-педагогічним складом, навчальними дисциплінами, системою оцінювання у вигляді рейтингових балів.

Таким чином здійснення локального моніторингу результатів успішності студентів дозволить визначати проблемні і сильні сторони навчального процесу групи, навчальної кафедри, факультету та надавати ґрунтовну інформації для визначення[17, 19]:

- потенціалу учнів та навчальних груп, відстеження їх прогресу впродовж періоду отримання освіти за визначеними спеціальностями відповідно до вимог Міністерства освіти;
- якості і рівня кваліфікованості науково-педагогічного складу, оцінка їх умінь і навичок доцільно і уміло використовувати методики викладання навчального матеріалу;
- факторів, які можуть вказувати на потенційні проблеми у питаннях академічної доброчесності і наявності корупційних дій, які негативно впливають на систему освіти.

Проведення комплексних моніторингових досліджень якості системи освіти в Україні є важливим і необхідним процесом, який надає необхідну інформацію стосовно стану системи освіти, відстеження змін у освітній траєкторії закладів вищої освіти та їх наслідків для держави на різних рівнях.

## 1.2 Критерії та показники для проведення моніторингу успішності

Використання аналітичних підходів у закладах вищої освіти є напрямом, який активно розвивається. Дослідження спрямовані на розвиток і використання навчальної аналітики почали розвиватися з 2010 року та набули широкої популярності під час пандемії Covid у 2019 році [9]. Перехід до дистанційного навчання створив нові виклики для академічної і наукової спільноти, які полягали у переході до навчання з використанням інформаційних систем для здійснення дистанційного навчання і виконання державного замовлення з підготовки кваліфікованих спеціалістами закладами освіти [14].

До викликів і проблем пов'язаних з дистанційним навчанням відносять [10]:

- зниження рівня соціалізації студентів та учнів через послаблення ролі викладача як агента соціалізації;
- неготовність здобувачами освіти опанувати навчальні матеріали самостійно;
- зростання випадків академічної недоброчесності і плагіату серед здобувачів освіти.

Дані проблеми і виклики створили усі необхідні умови для зниження рівня якості освіти в навчальних закладах України та об'єктивного оцінювання здобувачів освіти [11]. Враховуючи дані фактори з точки зору проведення локального моніторингу якості вищої освіти, результати успішності студентів є важливим і невід'ємним фактором, який присутній для різних форм навчання (очне, дистанційне, онлайн). Невід'ємним етапом проведення моніторингу результатів успішності є визначення показників і критеріїв, які необхідно визначати і обчислювати які корелюють з якістю навчання у вищих навчальних закладах [16, 21].

Критерій визначається як ознака, відповідно до якої здійснюється порівняльна оцінка об'єкту досліджується або конкретного явища. Його характеристики визначаються в залежності від цілей та предмету здійснення

моніторингу і він може бути розглянутий у вигляді допоміжного методу оцінки якості освіти [12, 16]. Під поняттям “показника” визначають міру визначення критерію, оскільки показники є доступними для спостереження і обліку, що в кінцевому результаті надає можливість надати комплексну оцінку об’єкту у відношенні до відповідного критерію.

У рамках проведеного дослідження за темою “Важливість розробки інформаційно-аналітичної системи для опрацювання та візуалізації результатів моніторингу успішності студентів” було визначено показники, які можуть бути використані для визначення і надання оцінки якості освіти у вищому навчальному закладі [3, 15]. Одним з основних показників проведення моніторингу результатів успішності є визначення середнього балу студентів, навчальних груп і факультетів впродовж усього періоду отримання вищої освіти. Значна відмінність у середній балах окремо взятого студенту між семестрами може вказувати на зміну його особистої мотивації у навчанні, пов’язану з методами навчання і викладання матеріалу різними викладачами, відмінність у поглядах студенту і викладача стосовно цілей і необхідності вивчення навчальної дисципліни у визначених обсягах.

Дослідження зміни середнього значення рейтингового балу у групах дозволяє робити припущення про мотивацію, якість і відповідальність викладачів під час проведення занять. Наприклад, коли викладачі стикаються з групами студентів які мають низький рівень знань, необхідний для опанування навчальної дисципліни у повному обсязі, рівень мотивації і якість викладання педагогічними працівниками може бути знижена та в кінцевому результаті впливати на об’єктивність підсумкового оцінювання [13].

У тих самих умовах, коли споріднені навчальні дисципліни проводить інший викладач, результати оцінювання можуть значно відмінюватись від першого випадку. Це може свідчити про неправильне використання підходів і методів навчання одним викладачем, на противагу високому професіоналізму іншого.

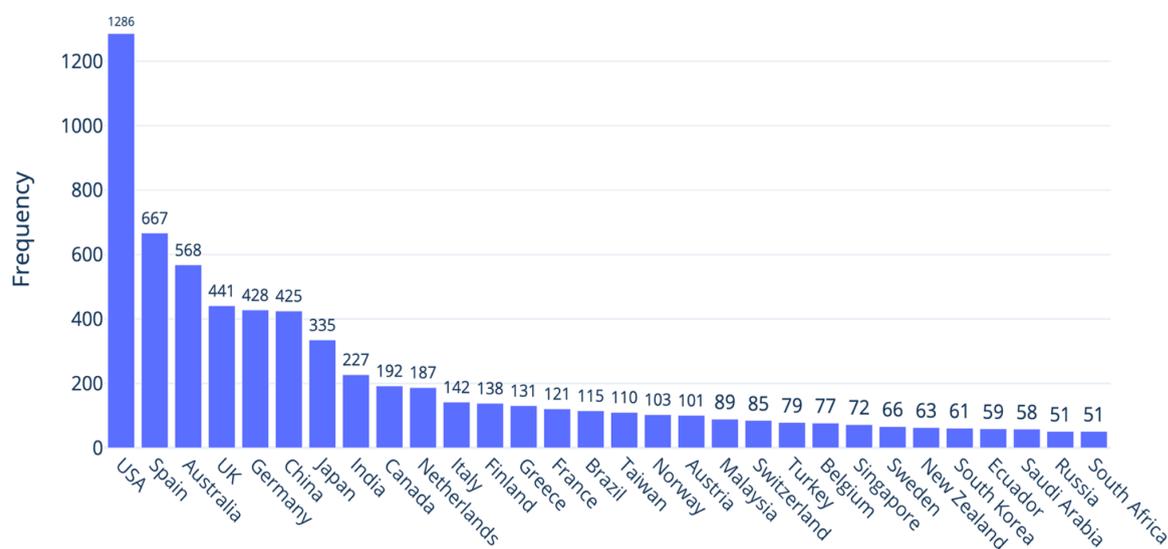
Визначення таких показників як медіана і рівень дисперсії (міра відхилення відносно середнього значення) мають більшу аналітичну значимість і цінність для виявлення потенційних ризиків порушення академічної доброчесності і корупції. Таким чином, одна з груп маючи велику різноманітність оцінок з більшості навчальних дисциплін, а з окремої – у навчальній групі різко змінилась їх пропорційність у вигляді високих балів. Даний випадок потребує подальшого аналізу, у порівнянні з результатами у інших навчальних групах. Якщо така тенденція присутня для груп, де навчальні заняття проводить один і той викладач, допустимо припускати, що він порушує академічну доброчесність. Для підтвердження або спростування даного припущення доцільно здійснити додаткові види моніторингу у проблемній ділянці освітнього процесу та проводити додаткові перевірки і розслідування.

Для великих груп студентів доцільно обчислювати стандартне відхилення, яке вказує на варіабельність оцінок навколо середнього арифметичного значення. У випадках коли значення стандартного відхилення є значним для різних груп, то це може вказувати на значні відмінності між навчальними групами та рівнем їх знань.

### **1.3 Навчальна аналітика в умовах цифровізації освіти**

Процес цифровізації освіти полягає не лише у збільшенні кількості сучасних технічних засобів, а й у зміні підходів до організації провадження освітньої діяльності, в тому числі процесу здійснення її моніторингу. Навчання, опосередковане технологіями, набуло помітного значення у вищій освіті, а навчальна аналітика стала потужним інструментом для відстеження взаємодії учнів у системі закладів вищої освіти. Використання навчальної аналітики дозволяє не лише виявляти слабкі місця в навчальному процесі, але й надає можливість здійснювати запобіжні заходи, які сприяють підвищенню ефективності навчання [15].

Здійснений огляд літератури пов'язаної з тематикою кваліфікаційної роботи надає підстави прийти до висновку, що в Україні не було проведено та задокументовано досліджень, які належним чином визначали б наявність і використання комп'ютеризованих систем для здійснення моніторингу у вищих навчальних закладах. Натомість, дослідження пов'язані зі створенням і впровадженням інформаційних систем для здійснення навчальної аналітики набирають популярності для високорозвинених країн [1]. Кількість наукових публікацій від науковців з різних країн, які містять у своїх назві поняття “навчальна аналітика” наведено на рис. 1.1.



(b)

**Рис. 1.1. Наукові публікації пов'язані з навчальною аналітикою**

*Джерело:* A Current Overview of the Use of Learning Analytics Dashboards. Режим доступу: <https://www.mdpi.com/2227-7102/14/1/82>

Спираючись на результати дослідження інших науковців, значна кількість наукових праць пов'язаних з навчальною аналітикою має теоретичний характер, допустимо вважати, що після проведення аналізу і формалізації їх результатів виникає необхідність практичної реалізації, експериментальне та фактичне впровадження інформаційних систем, які призначені для здійснення моніторингу та аналітики навчальної діяльності.

## **РОЗДІЛ 2. ВИБІР ТЕХНОЛОГІЙ І ВИКЛИКИ ТЕХНІЧНОЇ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ**

Використання інформаційних технологій і програмних продуктів необхідних для проведення моніторингу у закладах вищої є важливим і необхідним компонентом освітнього процесу. У XXI ст. існує широкий вибір технологій, архітектурних підходів і платформ для реалізації програмних продуктів. Широке використання десктопних інформаційних систем поступово поступило своє місце web-додаткам і мобільним платформам завдяки розвитку мережових технологій. У цей період використання web-технологій дозволяє здійснювати обробку даних і поширення інформації у реальному часі.

Використання інформаційних технологій у сфері освіти також сприятиме створенню нових, доступних та інтерактивних навчальних середовищ. Завдяки цьому студенти отримують можливість здійснення доступу до навчального матеріалу з будь-якого пристрою. Це дозволяє зробити навчання зручним і доступним з будь-якого місця.

Саме тому, реалізацію інформаційно-аналітичної системи для обробки, візуалізації результатів моніторингу результатів успішності студентів доцільно здійснювати у вигляді web-додатку.

### **2.1 Існуючі програмні засоби візуалізації статистичних даних**

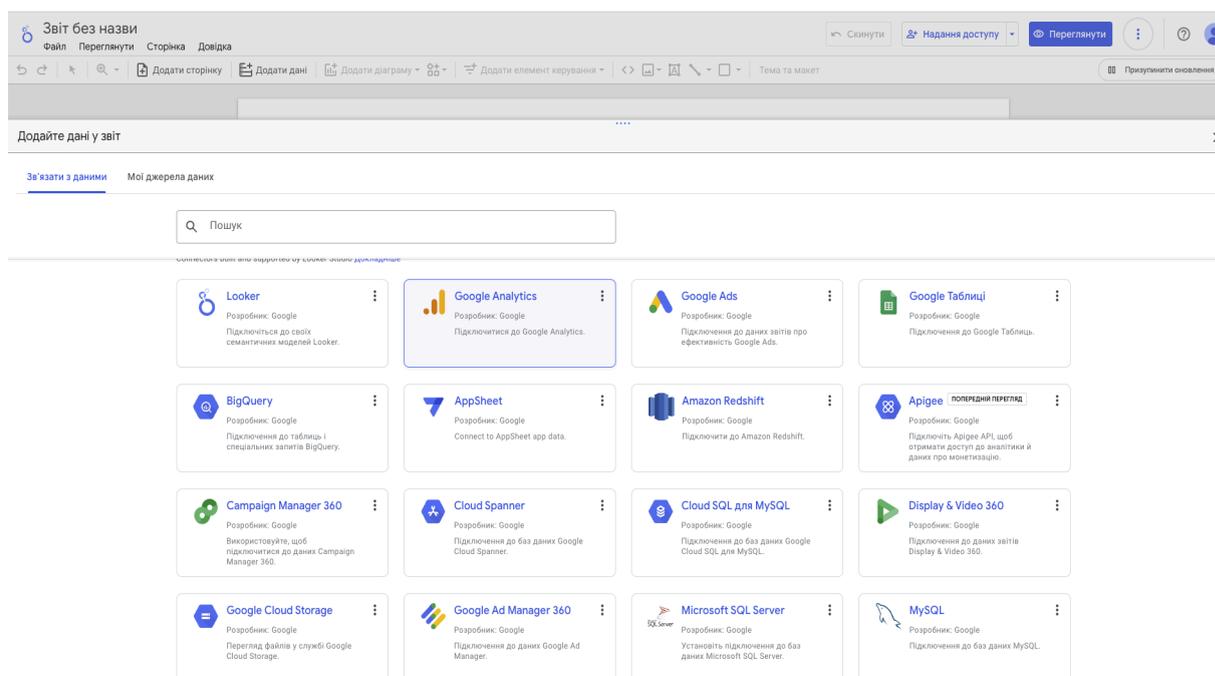
З фрагменту дослідження наведеного у третьому пункті першого розділу було визначено, що навчальна статистика є напрямом досліджень, який лише розвивається і проведені дослідження несуть в основному теоретичний характер.

Зважаючи на це, доцільним було провести аналіз існуючих програмних засобів для візуалізації статистичних даних з метою визначення функціональних особливостей, підходів, переваг та недоліком існуючих систем. Їх аналіз є доцільним тому, що використання реалізованих підходів до візуалізації

статистичних даних, які здобули широку популярність серед кінцевих користувачів свідчить про практичну ефективність даних систем.

Серед існуючих засобів візуалізації статистичних даних було обрано наступні web-додатки: “Looker Studio”, “Data Voh” та “Click Up”.

Looker Studio - платформа для аналізу та візуалізації даних, яка спрощує створення та розуміння звітів. Варто відзначити, що Looker Studio є удосконаленням існуючої системи Google Data Studio. Його головна перевага полягає в тому, що він дозволяє підключатися до джерел даних, таких як бази даних, електронні таблиці та інші програмні платформи створені компанією Google [21]. Доступні джерела даних наведені на рисунку 2.1.



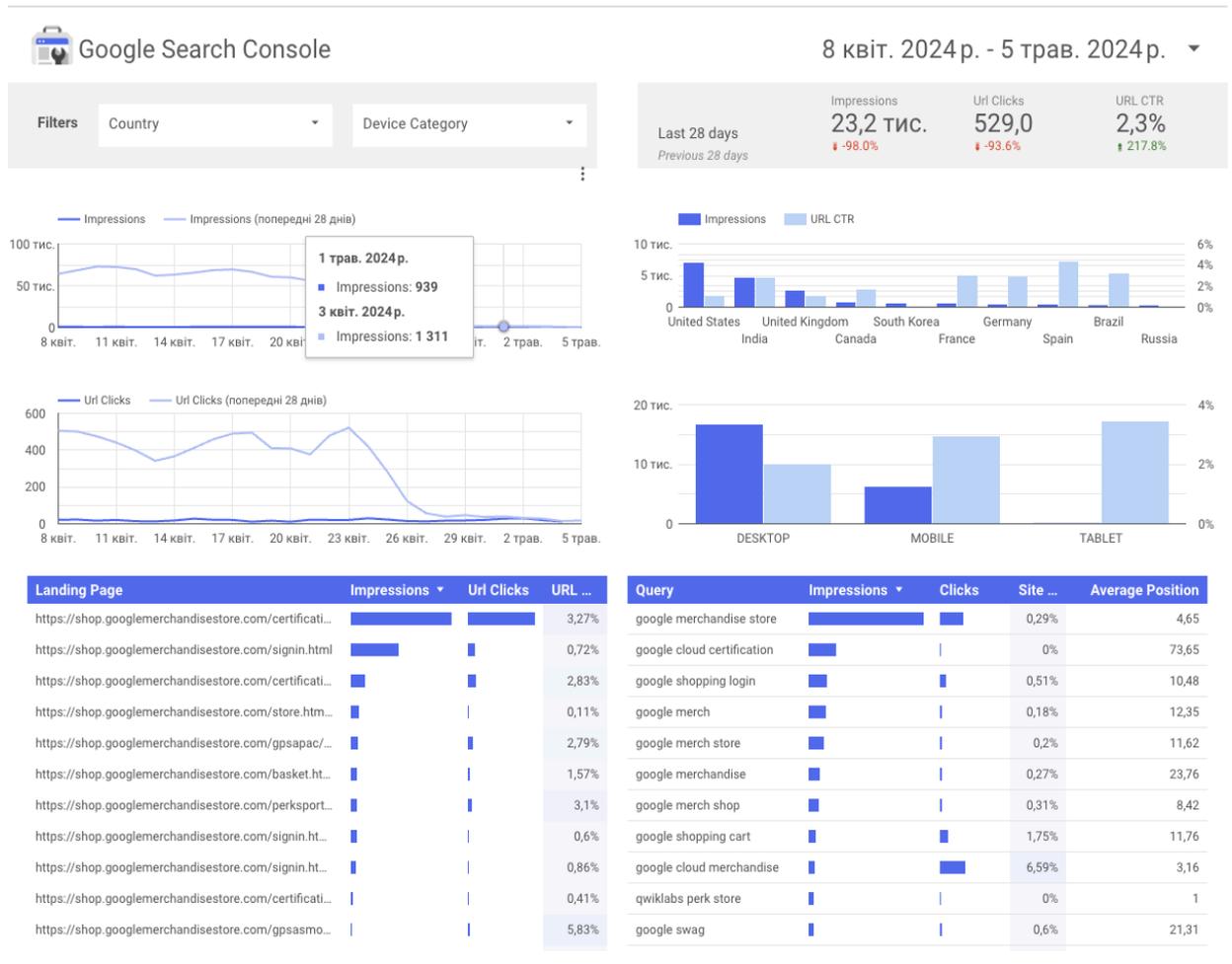
**Рис. 2.1 Доступні джерела даних Looker Studio**

Використання значної кількості джерел даних досягається завдяки інтеграції інтерфейсів взаємодії між різними програмними продуктами. Даний крок є необхідним для проведення і здійснення управління процесами всередині організації.

Основне призначення Looker Studio полягає у візуалізації статистичних даних і створенні інформаційних дошок для генерації електронних звітів, що є

корисними для здійснення моніторингу і прийняття управлінських бізнес рішень. Серед інструментів візуалізації Looker Studio містить значну кількість графіків та діаграм, які розміщуються на сторінці. Для графіків допустимо використовувати фільтри даних, для формування зручних і корисних звітів.

Приклад звіту з використанням інформаційних панелей і фільтрів даних зображено на рис. 2.2.



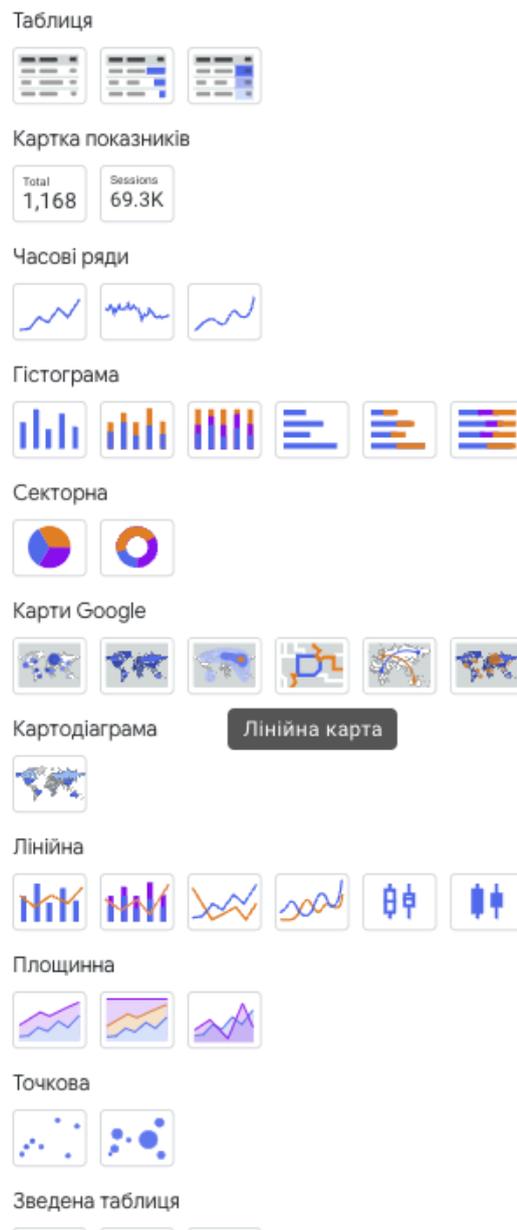
**Рис. 2.2** Звіт створений за допомогою використання Looker Studio

Загальний алгоритм використання Looker Studio має наступні етапи:

1. Визначення джерела даних – перший обов'язковий етап проведення моніторингу. Під час створення платформи компанією Google було передбачено, що здійснення моніторингу не може здійснюватись без необхідного набору даних.

2. Додавання графіків і діаграм на аркуш звіту. Для створення звіту використовується набір інтегрованих інструментів візуалізації, таких як Tableau від Salesforce.
3. Додавання фільтрів даних. Для формування зручних і гнучких звітів Looker Studio містить набір фільтрів даних, щоб спростити проведення кінцевого аналізу.

Перелік інструментів візуалізації даних відображено на рис. 2.3.



**Рис. 2.3 Компоненти візуалізації Looker Studio**

Як видно, компанія Google передбачала використання широкого інструментарію даних для різного спектру завдань візуалізації, проте виникає складність створення динамічних діаграм, призначених для вивчення та здійснення подальшого аналізу [24].

Підсумовуючи проведений аналіз функціональних особливостей програмного продукту Looker Studio, визначено наступні переваги та недоліки:

#### 1. Переваги:

- значна кількість доступних джерел даних;
- зручний інтерфейс, який надає необхідні інструменти для побудови інформативних звітів;
- широкий набір доступних діаграм, таблиць і карток для статистичних показників;
- наявність фільтрів для зміни параметрів візуалізації віджетів;
- інтеграція з різними існуючими програмними продуктами;
- можливість збереження історії звітів для здійснення системного моніторингу;
- активна підтримка компанією Google та її партнерами.

#### 2. Недоліки:

- орієнтація на здійснення моніторингу і аналізу даних комерційних установ і організацій, що користуються сервісами Google;
- фільтри призначені для визначення кількісних характеристик досліджуваного об'єкту за одним критерієм;
- візуалізація даних здійснюється за допомогою статичних віджетів.

Використання платформи Looker Studio є гарним рішенням для здійснення управлінського моніторингу у комерційних установах та маркетингових відділах. Її використання в умовах здійснення освітнього моніторингу результатів успішності студентів може вимагати постійної підтримки і оновлення для забезпечення працездатності.

Databox – це програмне забезпечення для візуалізації даних, розроблене та спеціалізоване для створення комплексних інформаційних панелей для бізнесу. Інструмент має два відповідних тарифних плани: безкоштовний – для персонального використання, за підпискою – для групового використання.

Завдяки великій бібліотеці готових шаблонів і інтеграції з популярними джерелами даних Databox дозволяє користувачам швидко збирати та налаштовувати панелі визначення ефективності і продуктивності. Формування звітів для аналізу дані за допомогою гістограм, точкових діаграм, лінійних діаграм та інших графіків візуальної аналітики [25].

Оновлення даних платформи в режимі реального часу та доступність мобільних пристроїв дозволяють командам легко контролювати свою продуктивність і приймати управлінські рішення.

Платформа автоматично генерує звіти у вигляді дашбордів з набором діаграм призначених для візуалізації і обчислення попередньо визначених показників. Джерела даних доступні у Databox наведені на рис. 2.4.

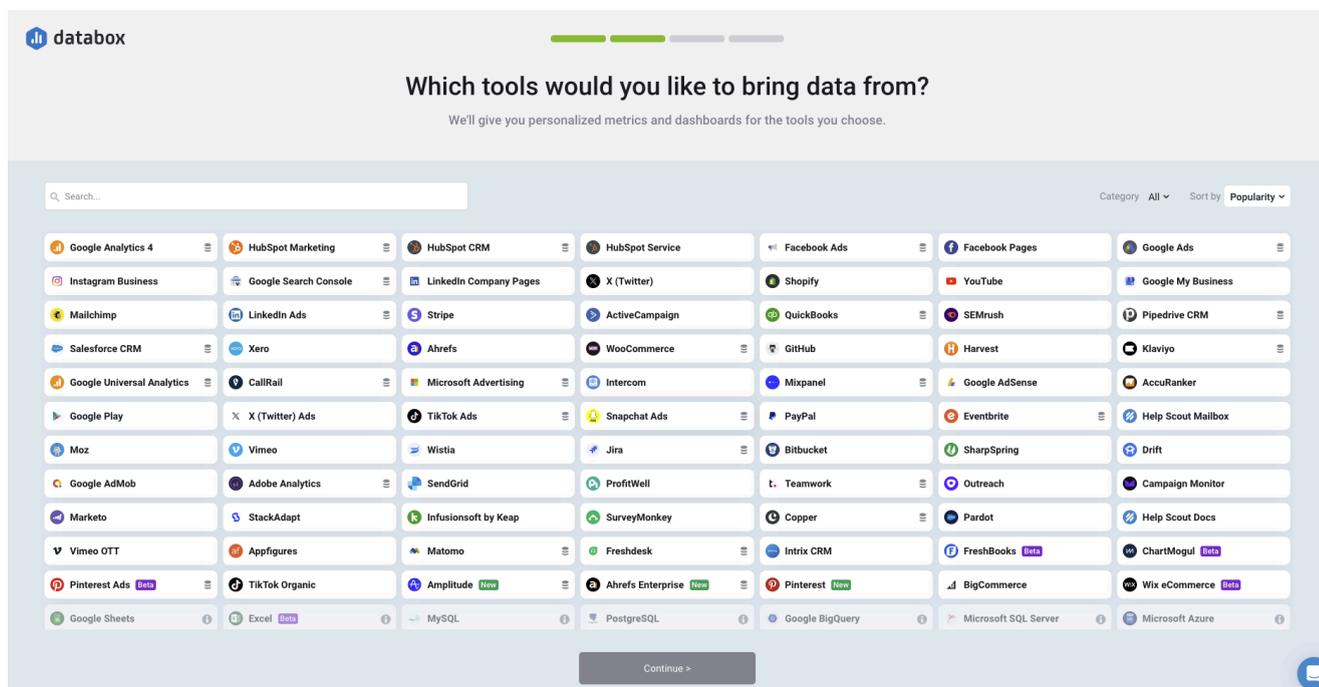


Рис. 2.4 Доступні платформи джерел даних для Databox

Специфічність даної платформи зумовлена тим, що спонсорами і реалізаторами даного програмного продукту є комерційні організації, які сфокусовані на розвитку і вдосконаленні свого бізнесу.

Дослідження даного продукту дозволяє дійти висновку, його що використання для здійснення моніторингу успішності у закладах вищої освіти є неможливим через наступні фактори:

1. Продукт є комерційним і його використання у межах навчального закладу вимагає оформлення підписки;
2. Джерелами даних можуть виступати лише дані, які є заздалегідь визначеними у Databox
3. Створення і формування звітів виконується автоматично, у визначених платформою алгоритмах, що є недостатнім для здійснення ефективного моніторингу у закладах освіти.

Click Up – уніфікована платформа, призначена для управління проектами та візуалізації даних, що дозволяє приймати обґрунтовані менеджерські рішення та здійснювати постійний моніторинг і аналіз продуктивності. Дана платформа є безкоштовною лише з обмеженою функціональністю і призначенням для персонального використання [26].

Для здійснення моніторингу діяльності установи у Click Up використовується широкий набір функціональних особливостей [27]:

- обрання широкого набору даних включаючи електронні таблиці і бази даних;
- можливість проведення документування процесів проведення і результатів моніторингу;
- набір діаграм для візуалізації даних;
- проведення колективної роботи кількома учасниками процесу здійснення моніторингу і управління;
- створення користувальницьких віджетів з користувальницькими параметрами.

Інтерфейс застосунку складається з меню, компоненти якого розділені за категоріями, а результати і компоненти за допомогою яких здійснюється моніторинг певних процесів відображені на дашборді з використанням елементів фільтрації даних (рис 2.5).

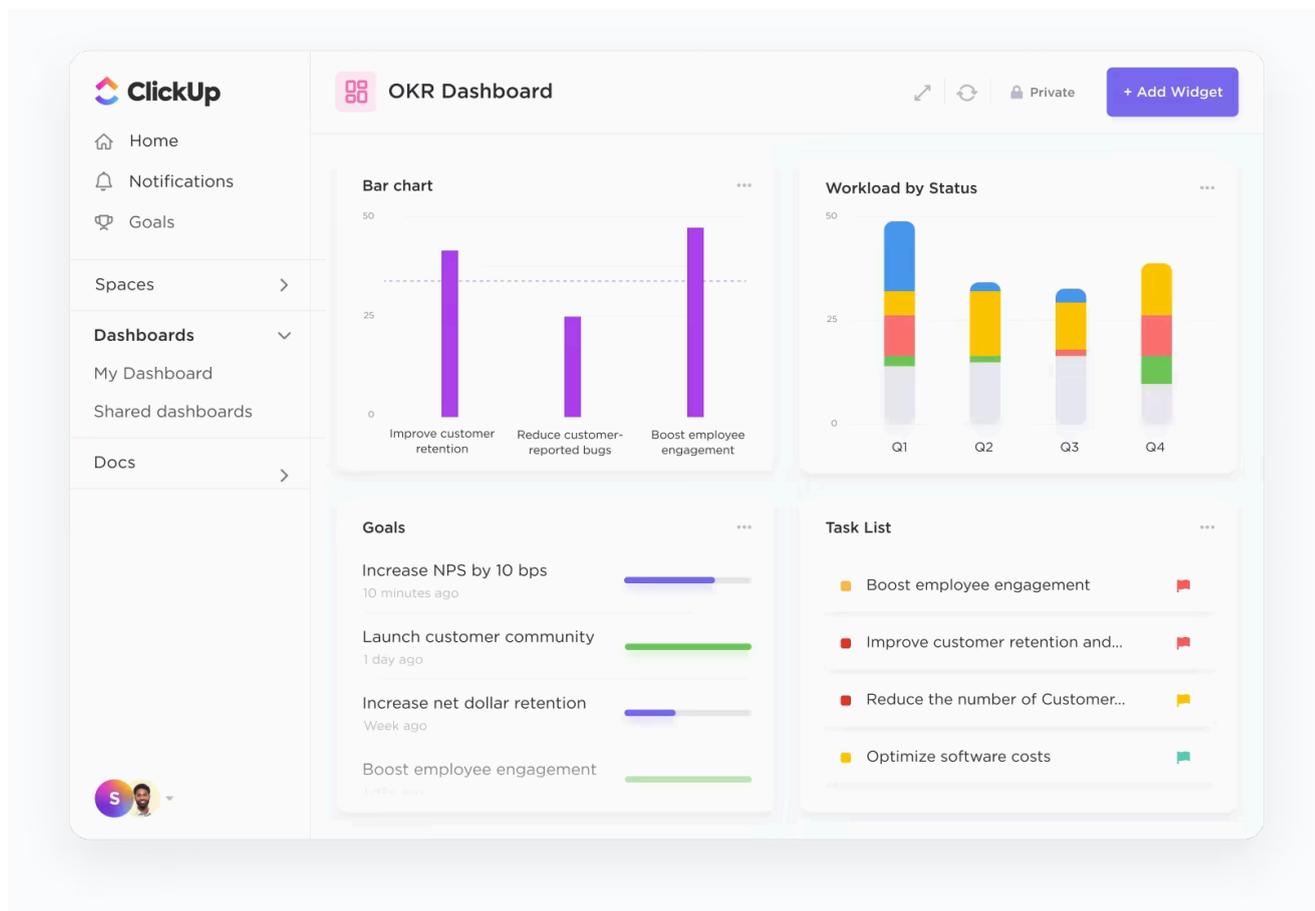


Рис. 2.5 Інтерфейс платформи Click Up

*Джерело:* The all-in-one platform for projects. Режим доступу:

<https://clickup.com/teams/project-management>

Після проведеного огляду функціональності web-платформи “Click Up” зроблено висновки стосовно особливостей, які можуть бути використані під час створення інформаційно-аналітичної системи для обробки та візуалізації результатів успішності студентів. Використання у закладах вищої освіти безкоштовної версії платформи, що була проаналізована, є неможливим через обмежену функціональність і обмеженість наданих ресурсів.

## 2.2 Особливості технічної реалізації

Дослідження особливостей здійснення моніторингу якості освіти у закладах вищої освіти, публікацій спрямованих на дослідження і розвиток напряму навчальної аналітики та існуючих засобів візуалізації статистичних даних надає необхідну інформацію для визначення особливостей технічної реалізації практичної частини кваліфікаційної роботи.

Створення програмних продуктів є послідовним процесом, який складається з визначеного набору етапів і фаз [26]. Успішність реалізації і використання інформаційно-аналітичних систем значною мірою залежить від дослідження цільових потреб користувачів, тенденцій і напрямів розвитку існуючих додатків. Враховуючи, що інформаційно-аналітична система буде застосовуватись для здійснення моніторингу закладу освіти необхідно врахувати наступні складові:

1. Визначення користувальницького сценарію використання системи.
2. Формування функціональних і системних вимог.
3. Обрання сучасних технологій, що дозволяють реалізувати необхідну функціональність.
4. Реалізація необхідних методів і сервісів призначених для збору, обробки і обчислення необхідної для моніторингу інформації.
5. Проектування і розробка компонентів інтерфейсу і візуалізації даних;
6. Проведення тестування на фазі активної розробки інформаційної системи, з метою скоротити часові витрати на виявлення і усунення дефектів.
7. Передбачити архітектурні особливості системи, що дозволяють здійснити інтеграцію з іншими web-додатками.

Виконання наведених робіт додатково доповнюється урахуванням переваг і недоліків досліджених платформ візуалізації статистичних даних, серед яких особливими викликами технічної реалізації є:

- використання технологій, які не потребують додаткових витрат;
- забезпечення можливості динамічної фільтрації даних на

компонентах візуалізації;

- забезпечення достовірного відображення наборів даних відповідно до обраних джерел даних.

Після визначення особливостей технічної реалізації необхідним є вивчення існуючих інструментів побудови вебзастосунків, призначених для опрацювання і візуалізації даних для реалізації інформаційно-аналітичної системи моніторингу успішності студентів.

### **2.3 Інструменти побудови вебзастосунків**

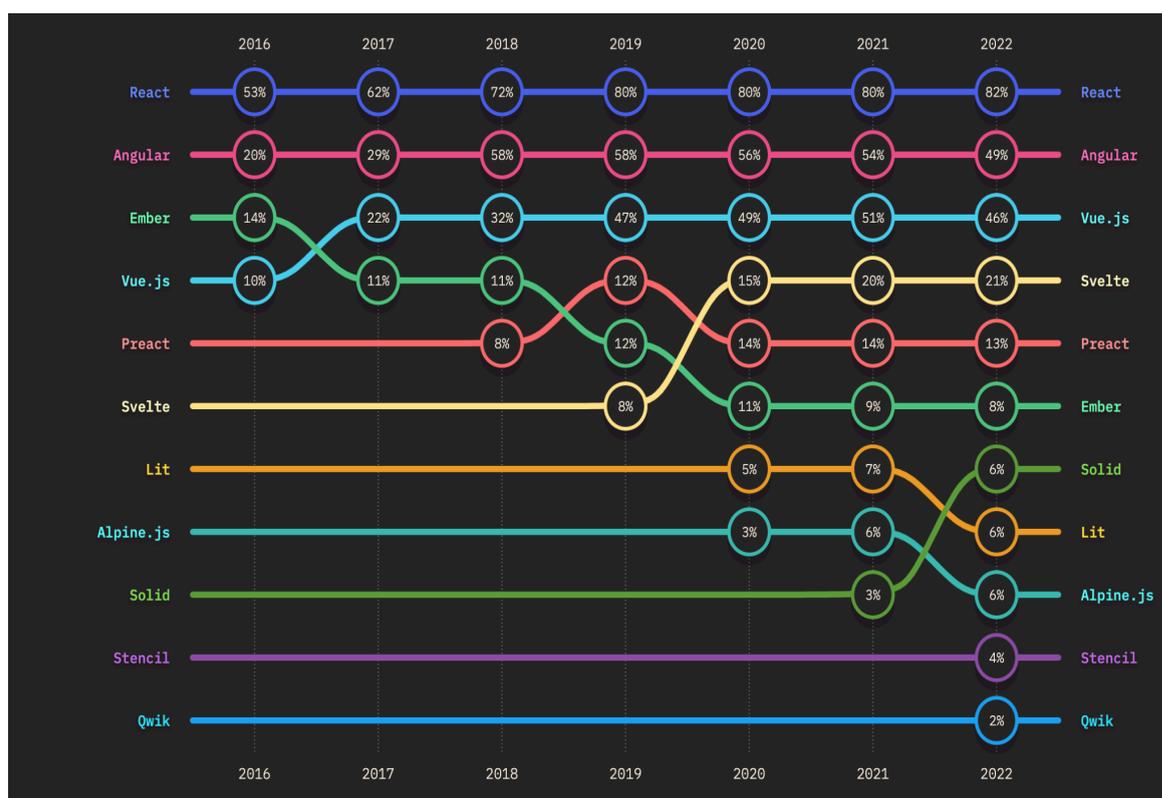
Стрімкий розвиток інформаційних технологій заклав додаткову необхідність у скороченні часових витрат для створення програмних продуктів. Це пов'язано з тим, що для бізнесу і підприємств важливо забезпечувати рішення для актуальних і нагальних потреб користувачів, щоб мати можливість залишатися конкурентоспроможними. З метою вирішення цієї проблеми створюються фреймворки - інструменти, що реалізують загальноприйняті архітектурні патерни і набори бібліотек для швидкої розробки програмних продуктів.

У світі web-розробки, зокрема для розробки серверних та клієнтських частин додатків активно використовується мова програмування JavaScript. Для реалізації інтерфейсу в браузерях JavaScript є чи не єдиним доступним рішенням. За допомогою HTML створюється розмітка вебсторінок та визначається їх структура, за допомогою CSS (англ. “cascading style sheets” - каскадні таблиці стилів), визначається стилістичне оформлення та анімування HTML елементів, а за допомогою JavaScript – реалізують програмну логіку.

Дана мова програмування має активну підтримку та постійно розвивається, стандартизується. Одним з найбільш популярних стандартів, що використовується є стандарт ECMAScript 2015, зокрема відомий як стандарт “ES6” [27]. Широка сфера використання, необхідність вирішення подібних

проблем та завдань призвела до виникнення великої кількості фреймворків, таких як Angular, React, Vue.js, Ember, Svelte та інші.

З метою визначення необхідного фронтенд фреймворку, серед наявних рішень, для створення інформаційно-аналітичної системи для опрацювання та аналізу результатів моніторингу успішності студентів було використано відомості щорічного опитування розробників щодо екосистеми JavaScript наведені на рисунку 2.6.



**Рис. 2.6** Рейтинг популярності JavaScript фреймворків у 2022 році

Джерело: State of JavaScript. Режим доступу: –

<https://2022.stateofjs.com/en-US/libraries/front-end-frameworks>

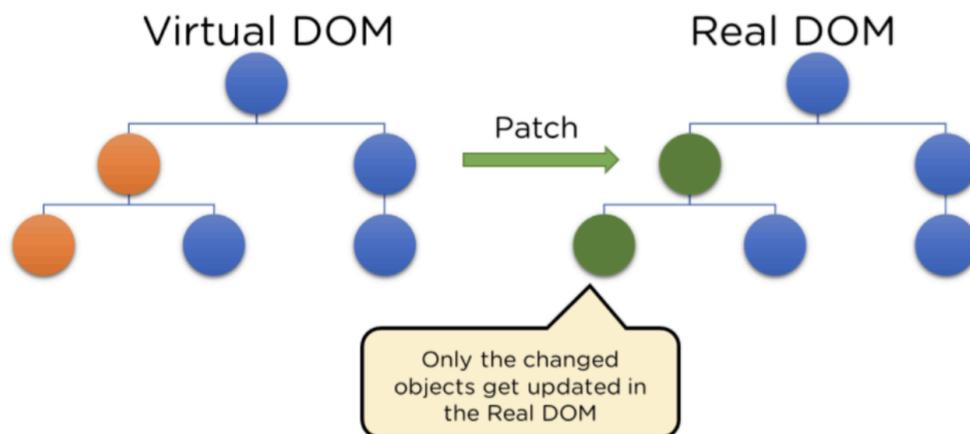
З метою визначення фреймворку, який є доцільним для виконання практичної частини дослідження було обрано три найбільш популярні. З результатів опитування видно, що найпопулярнішими рішеннями є React, Angular та Vue.js.

React - це бібліотека JavaScript з відкритим вихідним кодом, розроблена Facebook, яка використовується для створення інтерфейсів користувача або

компонентів інтерфейсу користувача [28]. Ця бібліотека вперше з'явилася в травні 2013 року і зараз є однією з найбільш часто використовуваних інтерфейсних бібліотек для веб-розробки. React пропонує різні розширення для підтримки всієї архітектури програми, наприклад Flux та React Native, крім простого інтерфейсу користувача.

Особливою відмінністю React, від інших фреймворків є використання спеціального синтаксису JSX, який дозволяє спільно використовувати HTML і Javascript. Також дана бібліотека використовує спеціальний віртуальний DOM (англ. “document object model” - об'єктна модель документа) – це спеціальний підхід у програмуванні, який полягає у співставленні реального і віртуального DOM. Даний підхід використовується з метою реалізації динамічного оновлення елементів HTML сторінки.

Схематичне зображення підходу з використанням віртуального DOM наведено на рисунку 2.7.



**Рис. 2.7** Принцип віртуальної об'єктної моделі документа

*Джерело:* The Best Guide to Know What Is React. Режим доступу: –

<https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>

Взаємодія між віртуальною та реальною об'єктною моделями відбувається за наступним алгоритмом:

1. Внесення змін стану елементу у віртуальній моделі;
2. Оновлення дерева елементів віртуальної моделі;

3. Оновлення у реальній об'єктній моделі документа лише тих структурних елементів, що були змінені.

Архітектурні особливості React мають низку переваг при побудові маленьких і середніх web-додатків. Серед переваг можна визначити:

- Висока продуктивність. React використовує віртуальну DOM (модель об'єктів документа), що робить його швидшим, ніж традиційні бібліотеки JavaScript. Віртуальний DOM дозволяє React оновлювати лише необхідні компоненти, а не оновлювати всю сторінку, що призводить до швидкого часу візуалізації та покращення продуктивності.
- Багаторазові компоненти. Перевикористання існуючих повторюваних компонентів дозволяє зменшити кодову базу та спростити підтримку існуючого проєкту.
- Розширення DevTools. Розроблене розширення для браузера Google Chrome дозволяє проводити тестування продуктивності React компонентів безпосередньо у браузері.
- Підтримка. Популярність використання створило активну аудиторію розробників, які зацікавлені у розвитку і вдосконаленні існуючого інструменту.

Незважаючи на перелічені переваги бібліотека має ряд недоліків. Серед них виокремлюється використання спеціального синтаксису JSX створює проблеми для початківців, які крім базових знань повинні вивчити новий синтаксис і засвоїти компонентний підхід створення вебдодатків. Також використання віртуальної об'єктної моделі документа стає недоліком для додатків, які мають велику кількість компонентів, що оновлюються. Це створює проблеми продуктивності і працездатності таких продуктів.

Варто відзначити, що для реалізації повнофункціонального додатку є необхідність пошуку і інтеграції додаткових модулів і бібліотек у проєкт, що може створювати проблеми на етапі підтримки програмного забезпечення.

Архітектура фреймворку Vue.js, аналогічно до бібліотеки React,

використовує компонентно-орієнтований підхід. Вона заснована на концепції компонентів, які є незалежними блоками коду, що містять шаблони HTML, логіку JavaScript і стилі CSS. Ці компоненти можуть повторно використовуватися і взаємодіяти один з одним. Унікальною особливістю Vue.js виділяють його прогресивність. Вона полягає у тому, щоб використовувати лише ту функціональність, яка необхідна для виконання тих чи інших завдань без необхідності реалізовувати лише один архітектурний підхід.

Фреймворк має свої механізми управління станом компонентів, такі як Vuex – він надає централізоване сховище даних, до якого можуть звертатися різні компоненти застосунку. Це дає змогу легко керувати та синхронізувати стан web-сторінок. Vue також має інтеграцію з іншими інструментами та бібліотеками, наприклад Vue Router призначений для управління маршрутизацією та Vue CLI для швидкого налаштування проєкту.

Серед переваг і недоліків Vue.js визначають наступне [29]:

#### 1. Переваги:

- Легкість. Наявність базових знань у веброботці дозволяє створювати повнофункціональні додатки у короткі терміни.
- Компонентно-орієнтований підхід. Він реалізований аналогічно як у бібліотеці React.
- Використання нативного синтаксису Javascript.
- Підтримка та модульність. Завдяки аудиторії розробників, які зацікавлені у розвитку і вдосконаленні існуючого інструменту створено багато модулів і бібліотек які легко інтегруються у Vue додатки.

#### 2. Недоліки Vue.js:

- Відсутність масштабування. Відомі спонсори не підтримують фреймворк, а волонтерського фінансування на забезпечення стабільності і вбудованої функціональності для використання у масштабних проєктах недостатньо.

- Надмірна гнучкість коду. Вона призводить до збільшення нерегулярності та помилок у коді.
- Надмірно часті оновлення. Відсутність висококваліфікованих експертів у команді розробки призводить до того, що виправлені помилки у одній частині призводять до появи нових дефектів у іншій частині фреймворку.
- Зв'язування даних. При двосторонньому зв'язуванні повторно рендерить тільки ті частини коду, які були викликані і часто допускають помилки при читанні.

Використання фреймворку Vue.js дозволяє будувати гнучкі вебдодатки без наявності глибоких знань, проте використання у складних проєктах несе великі ризики для життєздатності системи під час використання таких систем.

Ангуляр – другий за популярністю фронтенд фреймворк, який активно підтримується компанією Google. Даний інструмент набув своєї значної популярності завдяки своїй повноцінності. Angular вважається повноцінною платформою для створення SPA (англ. “single page application” - односторінковий додаток). Односторінковим додатком прийнято вважати веб-додатки або веб-сайти, які динамічно взаємодіють з користувачем, переписуючи поточну сторінку замість завантаження нових сторінок [30].

Angular включає в себе такі основні складові:

1. Модулі - особливі структурні компоненти призначені для розподілення програмного додатку у вигляді логічно незалежних частин.
2. Директиви - спеціальні класи, використання яких надає додаткову поведінку елементів.
3. Сервіси - класи, використання яких дозволяє керувати функціональністю компонентів.
4. Ін'єкції залежностей - концепція Angular, що дозволяє налаштовувати залежності між компонентами.

## 5. Шаблони - схеми для певних фрагментів інтерфейсу користувача.

Особливою відмінністю Angular від інших фреймворків є те, що Angular здійснює маніпуляції з реальною об'єктною моделлю документу. Це дозволяє вирішити проблему побудови динамічних і складних компонентів візуалізації.

Однією з основних переваг фреймворку є використання усіх необхідних інструментів без необхідності встановлення додаткових модулів, на відміну від React та Vue.js. Також наявність однієї кодової бази забезпечує додаткову гнучкість під час розробки додатку, а Angular CLI дозволяє спростити написання коду та дебагу у режимі реального часу. За допомогою інтегрованого серверу, програміст має можливість відразу бачити зміни на локальному сервері та одразу виправляти недоліки без додаткових витрат часу.

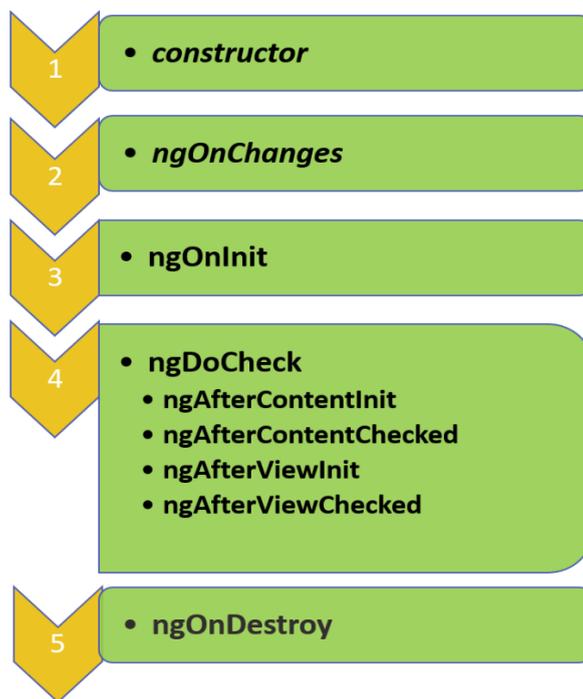
Директиви призначені для керування станом елементів і надають можливість динамічного створення, видалення або зміни елементів на основі умов або подій. У Angular міститься набір вбудованих директив, що можуть бути використані для побудови складних діаграм у інформаційно-аналітичних системах.

Наприклад, директиви `ngFor` або `ngIf`, дозволяють виводити дані в шаблоні, управляючи ітерацією списків або відображенням елементів на основі умов. Директиви `onClick`, `onChange`, `onInit` можуть реагувати на різноманітні події, такі як клік, введення даних або завантаження сторінки, і виконувати відповідні дії.

Залежно від конкретних завдань системи визначається метод зв'язування між компонентами, оскільки неправильно обрана форма зв'язування між компонентами створює суттєві труднощі під час розробки і реалізації динамічних інформаційно-аналітичних систем і, як наслідок, викликає нові ризики у коректності функціональної працездатності застосунку.

Також варто відзначити, що фреймворк надає гнучкість для керування станом компонентів не лише за допомогою директив, але й з використанням

життєвого циклу компонентів. Він складається з етапів, наведених на рисунку 2.8.



**Рис. 2.8** Етапи життєвого циклу Angular компонентів

Джерело: Angular Component Lifecycle. Режим доступу: –  
<https://geeksarray.com/blog/angular-component-lifecycle>

Як зазначено на малюнку, кожен з етапів життєвого циклу проходить у певній послідовності. При зміні стану компоненту Ангуляр виконуються події, визначені розробниками продукту. Ці хуки реалізовані в компонентах Angular за допомогою попередньо визначеного інтерфейсу, який повідомляє компоненту, яку подію ініціювати та коли автоматично викликати його метод.

Кожна з фаз має свої особливості використання:

1. **Constructor** – перший етап життєвого циклу компоненту. На даному етапі відбувається ініціалізація компоненту і визначення його стану за замовчуванням.
2. **ngOnChanges** – цей метод є хуком життєвого циклу в Angular, який викликається щоразу, коли відбувається зміна в одній із вхідних властивостей компонента.

3. `ngOnInit` – цей метод є хуком життєвого циклу, який викликається один раз після методу `ngOnChanges` і викликається, коли компонент ініціалізовано. За допомогою `ngOnInit()` ініціалізується логіка роботи самого компонента.
4. `ngDoCheck` – цей метод життєвого циклу в Angular, і він викликається, при детекції змін даного компонента. Це дозволяє реалізувати власний алгоритм виявлення змін для даного компонента. Він включає в себе додаткові перевірки, наприклад `ngAfterContentInit`, `ngAfterViewInit` та інші.

Багатокомпонентний життєвий цикл Angular компонентів надає широкі можливості для знаходження балансу між продуктивністю та функціональністю компонентів.

Дослідивши існуючі інструменти побудови клієнтської частини вебзастосунків вибір фреймворку Angular є найкращим рішенням для побудови інформаційно-аналітичної системи для обробки і візуалізації результатів успішності студентів, оскільки використання однієї кодової бази, директиви та хуки надають усі необхідні можливості для реалізації функціональності пов'язаної зі зміною компонентів візуалізації в залежності від користувальницьких фільтрів.

## РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ЗАСТОСУНКУ

### 3.1 Формування вимог до інформаційно-аналітичної системи

Для створення програмного додатку формування вимог є обов'язковим етапом розробки, спрямованим на задоволення потреб проблемної області та користувачів. Вимоги в свою чергу поділяються на значну кількість категорій, зокрема основними з них виділяють [31]:

1. Бізнес вимоги – це вимоги високого рівня, які виражають цілі й бажані результати організації. У формулюванні бізнес вимог визначається, які проблеми у певній проблемній області має вирішувати певний програмний продукт.
2. Користувальницькі вимоги – забезпечують визначення того, що система повинна робити для своїх користувачів, щоб у свою чергу відповідати бізнес-цілям. В свою чергу поділяються на:
  - Функціональні вимоги – призначені для опису роботи користувача з додатком. Як правило вони створюються на основі сценаріїв використання (англ. “use case”) або історій користувача (англ. “user story”).
  - Нефункціональні вимоги – визначають відповідність програмного забезпечення до законодавчих вимог, міжнародних специфікацій і стандартів.
3. Системні вимоги – визначають перелік необхідних апаратних характеристик, додаткових програмних засобів і платформ необхідних для забезпечення працездатності програмного продукту.

Визначення вимог є важливим і невід'ємним етапом створення програмного забезпечення. Даний етап здійснюється після проведення аналізу проблемної області і визначення кінцевої мети інформаційного продукту, що розроблятиметься.

У першому розділі кваліфікаційної роботи було проведено аналіз

особливостей здійснення локального моніторингу у закладах вищої освіти. Його результати дозволяють сформулювати наступні вимоги високого рівня для інформаційно-аналітичної системи для обробки і візуалізації результатів моніторингу успішності студентів:

1. забезпечити обробку результатів екзаменаційної сесії з електронних таблиць для використання даних у моніторингу і проведенні аналізу успішності навчальних груп;

2. візуалізувати опрацьовані результати успішності студентів за допомогою інформаційних панелей візуалізації з метою проведення подальшого аналізу;

3. забезпечити інтуїтивно зрозумілий для користувача інтерфейс з метою запобігання необачних дій, які становлять загрозу для працездатності системи.

Дані вимоги визначають глобальні завдання, що постають перед інформаційною системою. Виконання цих вимог дозволить використовувати застосунок у навчальних закладах для здійснення моніторингу з метою вдосконалення навчального процесу.

Для створення ефективної системи, яка може бути використана як технічний засіб здійснення моніторингу для проведення подальшого аналізу візуалізованих результатів успішності здобувачів вищої освіти реалізовано сценарій використання системи, який складається з наступних етапів:

1. Користувач повинен завантажити таблиці з розширенням .xls, .xlsx у яких містяться результати успішності студентів за підсумками екзаменаційної сесії. Приклад таблиці наведений у додатку А.
2. Після завантаження даних користувач має можливість переключення між трьома рівнями представлення даних: результати успішності кафедри, групи, окремого студента.
3. Використати результати обробки результатів успішності студентів з метою здійснення подальшого аналізу освітньої діяльності у навчальних групах.

Також передбачено використання альтернативних шляхів використання:

1. Наявність використання фільтрів надає можливість динамічної зміни

компонентів представлення у відповідності до заданих правил.

2. При виявленні проблем у роботі системи користувач має можливість зв'язатися зі службою підтримки та/або розробником системи.
3. Забезпечення інформаційної підтримки з описом використання системи

Реалізація сценарію використання системи та його альтернативних шляхів надає перевагу для знаходження і виправлення помилок впродовж періоду використання застосунку користувачами. Також, наявність зворотнього зв'язку між користувачами та розробниками відкриває перспективи для вдосконалення існуючої системи шляхом додавання нових функціональних можливостей, які можуть бути зумовлені зміною підходів до здійснення освітнього моніторингу.

У третьому підпункті третього пункту другого розділу кваліфікаційної роботи було визначено, що фреймворк Angular є найкращим рішенням для проектування клієнтської частини інформаційної системи для обробки і візуалізації результатів моніторингу успішності студентів. Ангуляр використовує платформу Node.js для виконання JavaScript коду на сервері.

Node.js – це кросплатформне середовище виконання JavaScript із відкритим кодом. Node.js запускає двигун V8 JavaScript, ядро Google Chrome, поза браузером. Це дозволяє Node.js бути дуже продуктивним [32]. Зважаючи на це, для використання застосунку накладаються додаткові системні вимоги, які визначаються не лише версією операційної системи, але й платформою Node.js. Системні вимоги до системи визначені наявністю програмного для двох оточень: на сервері та на пристроях розробника.

Для здійснення розробки і оновлення програмного забезпечення використання наступного програмного забезпечення є необхідним:

1. Операційна система: Mac OS/Linux/Windows 10/Windows 11
2. Платформа Node.js v18.20.3 і вище.
3. Пакетний менеджер npm v10.5.1. Призначений для встановлення залежностей двох типів: пакети необхідні для роботи програмного продукту, залежності необхідні лише на етапі розробки.

4. Angular v16.1.0.
5. Chart.js v4.4.1 – бібліотека призначена для побудови графіків та діаграм візуалізації даних.
6. Angular Material v16.2.13 – бібліотека компонентів інтерфейсу для використання у Ангуляр додатках.
7. Додаткові залежності, які можуть знадобитись розробникам на етапі створення застосунку.

Дані системні вимоги накладаються лише на розробників на етапі створення і підтримки системи. Додаток може бути розміщений на серверах навчального закладу, з операційними системами, які здатні використовувати платформу Node.js та пакетний менеджер npm, який встановить необхідні залежності у проєкт.

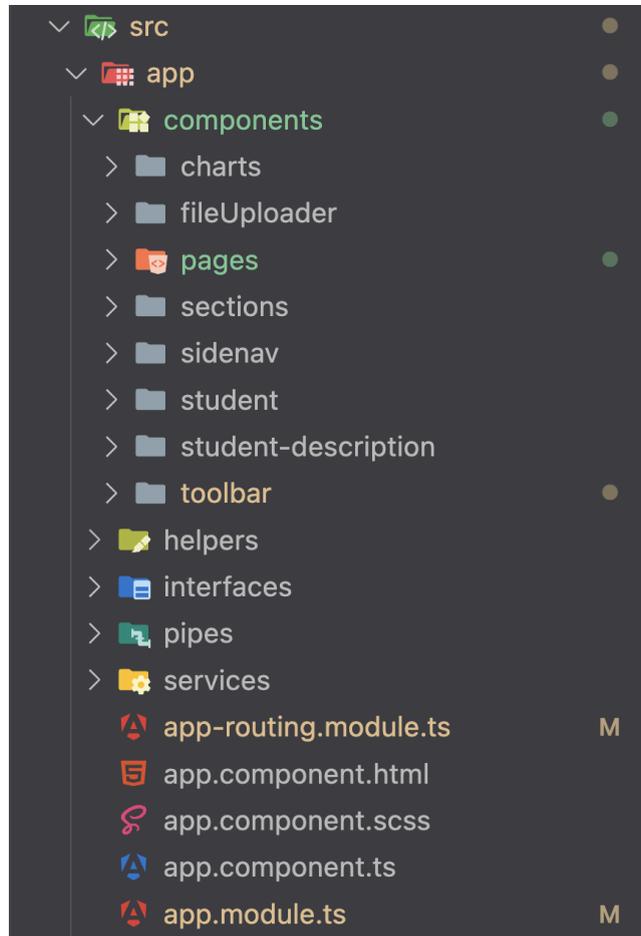
Користувачу достатньо мати на персональному комп'ютері один з сучасних браузерів, такі як Google Chrome, Safari, Mozilla Firefox та інші.

Реалізація і дотримання визначених системних, функціональних і бізнес вимог дозволяє розробити інформаційно-аналітичну систему для обробки та візуалізації результатів успішності студентів у відповідності до потреб закладів вищої освіти з урахуванням потреб користувачів системи. За рахунок використання даного архітектурного підходу користувачі не мають потреби встановлення додаткового програмного забезпечення на свої пристрої. Відсутність залежностей у роботі з пристроями користувачів є перевагою, оскільки це мінімізує кількість проблем і помилок роботи системи пов'язаних з конфігурацією оточення на стороні клієнта.

### **3.2 Практична реалізація системи**

Першим етапом створення інформаційно-аналітичної системи для опрацювання та візуалізації результатів моніторингу успішності студентів є визначення архітектури і файлової структури проєкту. Використовуючи Angular CLI (англ. “command line interface” – інтерфейс командної стрічки) згенеровано

архітектуру додатку, що використовує компонентний підхід до реалізації односторінкових клієнтських додатків. Базову файлову структуру визначено на рисунку 3.1.



**Рис. 3.1** Файлова структура базового додатку Angular

У директорії з відносним шляхом “./src/app” розташовуються усі необхідні компоненти додатку, розподілені за своїм призначенням:

1. Components – містить в собі компоненти інтерфейсу. Компоненти розподілені за логічним призначенням по директоріях.
2. Helpers – набір допоміжних класів для обробки даних і обчислень.
3. Interfaces – набір інтерфейсів, що описують структури даних. Забезпечують додаткову валідацію об’єктів з якими взаємодіють компоненти додатку.
4. Pipes – набір директив, які призначені для визначення користувальницьких фільтрів відображення даних

5. *Services* – містить сервіси призначені для обробки результатів успішності студентів та допоміжні сервіси для забезпечення взаємодії між компонентами.

У файлі “*app-routing.module.ts*” визначено маршрути, згідно яких користувач здійснює навігацію у додатку. Лістинг програмного коду файлу маршрутизації наведено на рисунку 3.2.

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
// components      You, 2 months ago • finalized student section
import { StudentStatsComponent } from './components/sections/student-stats/student-stats.component';
import { StudentComponent } from './components/student/student.component';
import { FileUploaderComponent } from './components/fileUploader/file-uploader.component';
import { GroupsComponent } from './components/sections/groups/groups.component';
import { DepartmentsStatsComponent } from './components/sections/departments-stats/departments-stats.component';
import { AboutComponent } from './components/pages/about/about.component';
import { DocsComponent } from './components/pages/docs/docs.component';
import { ContactsComponent } from './components/pages/contacts/contacts.component';

const routes: Routes = [
  { path: 'students', component: StudentStatsComponent },
  { path: 'groups', component: GroupsComponent },
  { path: 'uploadData', component: FileUploaderComponent },
  { path: 'student', component: StudentComponent },
  { path: 'department', component: DepartmentsStatsComponent },
  { path: 'about', component: AboutComponent },
  { path: 'docs', component: DocsComponent },
  { path: 'contacts', component: ContactsComponent },
];

You, 2 months ago | 2 authors (Mykhailo Kosinov and others)
@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule],
})
export class AppRoutingModule {}
```

**Рис. 3.2** Лістинг файлу маршрутизації додатку

Використання динамічної маршрутизації забезпечує навігацію між модулями функціонального і інформаційного меню, без відправки додаткових запитів до серверу.

Другим етапом реалізації системи є визначення її структурних і логічних компонентів, з якими користувач здійснює взаємодію. З метою реалізації функціональних вимог додаток розподілено на три секції:

1. Студенти – містить відомості про успішність кожного здобувача освіти та фільтри: пошук студента за прізвищем, пошук студентів за

групою, фільтр відображення освітніх досягнень за типом підсумкового контролю( залік, екзамен).

2. Групи – включає в себе інформаційні панелі візуалізації результатів успішності навчальних груп. Надає інформацію про рейтинг студентів у групі, динаміку успішності навчальної групи за семестрами, кільцева діаграма з розподілом навчальних балів студентів за рівнями знань, інформаційні картки зі значеннями статистичних показників.
3. Дашборд – здійснює обробку і візуалізації результатів успішності студентів навчальної кафедри.

Третім етапом створення інформаційно-аналітичної системи візуалізації є реалізації користувальницького сценарію враховуючи досвід популярних платформ візуалізації даних, згідно якого, користувач не має можливостей для перегляду вмісту секцій до завантаження даних, які візуалізуються.

Коли користувач взаємодіє з застосунком, передбачено відображення інформаційного повідомлення про необхідність завантаження даних. У функціональному меню міститься елемент “Додати дані”. Користувач виконавши клік по заданому елементу переходить до компоненту, який здійснює завантаження файлу даних у лише визначеному форматі .xls та .xlsx. Клік на елемент “Обрати файл для завантаження” відкриває файловий провідник пристрою користувача і виділяє доступні для завантаження розширення файлів.

Визначення форматів файлів доступних для завантаження запобігає неправильному використанню системи, внаслідок чого дані не можуть бути опрацьовані та візуалізовані. Після завантаження даних користувач має можливість перегляду оброблених даних відповідно до кожної секції.

Секція “Студенти” реалізована у вигляді картки для кожного студента. У ній передбачено наступні функціональні можливості:

1. Фільтри
2. Компонент “Pagination” – визначає кількість видимих карток з даними

студентів на сторінці.

### 3. Картки студентів

На рисунку 3.3 представлена картка студента.

Степанець Адріана		KN-960 навчальна група, 3 семестр навчання				
<input checked="" type="checkbox"/> Екзамени <input type="checkbox"/> Заліки						
№	Дисципліна	Рейтинг. бал	Тип контролю	Семестр	Викладач	Дата
1	WUQA	81	екзамен	1	Марія Доброславівна Сплюха	10.10.2022
2	JJVD	79	екзамен	1	Яловий Буйтур	24.7.2023
3	QJXC	65	екзамен	2	Паращук Лідія	24.9.2023
4	KAJU	73	екзамен	2	Дана Ющик	19.3.2023
5	WHRV	87	екзамен	2	Скоропадська В'ячеслава	21.8.2022
6	XVRI	97	екзамен	3	Болеслава Кандиба	24.5.2022
7	XVDR	92	екзамен	3	Андрухович Інна	24.12.2022
8	UUGF	72	екзамен	3	Тамара Адамівна Гордійчук	18.12.2023

**Рис. 3.3 Картка студента у секції “Студенти”**

Картка студента складається з наступних елементів:

1. Прізвище, ім'я студента
2. Навчальна група та поточний семестр навчання
3. Фільтр відображення за типом контролю
4. Таблиця з результатами успішності за результатами екзаменаційної сесії.

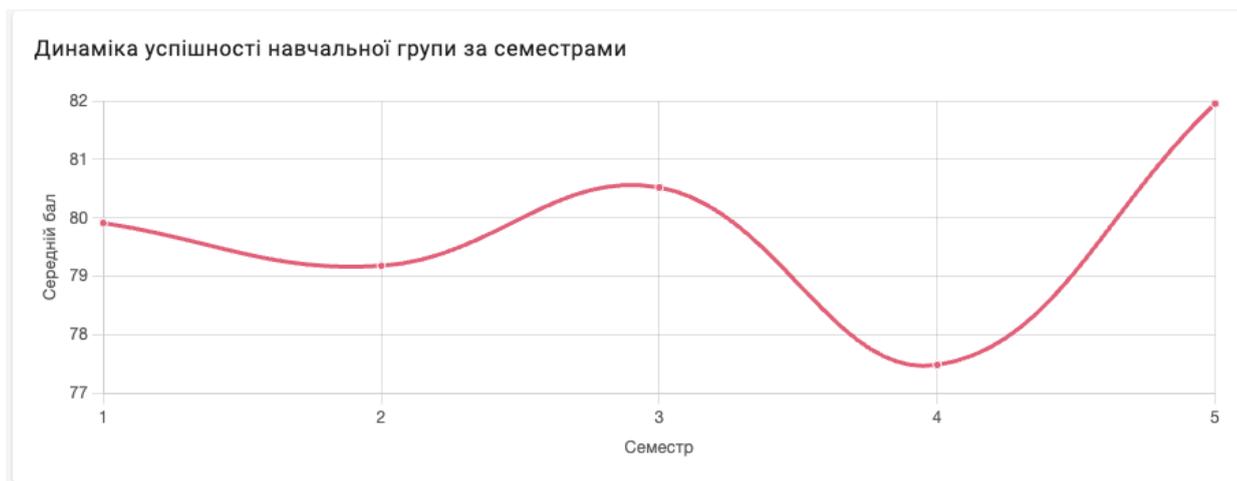
Таблиця відображає дані з назвою навчальної дисципліни, рейтинговим бал за національною шкалою, тип підсумкового контролю, відомості про викладача, який здійснював оцінювання, дата проведення підсумкового контролю з навчальної дисципліни.

За допомогою фільтру користувач має можливість перегляду успішності студенту за певним видос підсумкового контролю. Коли обрано “екзамени” та “заліки”, то для студенту відображаються усі відомості успішності за період навчання. Для зручності використання вони відсортовані за семестрами.

Секція “Групи” містить відомості про успішність навчальної групи за

результатами проведення останнього екзаменаційного контролю. Дана секція складається з наступних компонентів:

1. Фільтр відображення даних за назвою навчальної групи;
2. Статистичні показники групи;
3. Інформаційна панель “Динаміка успішності навчальної групи за семестрами” (рис. 3.4);



**Рис. 3.4 Візуалізація динаміки успішності навчальної групи у кожному семестрі**

4. Інформаційна панель розподілу балів студентів за рівнем знань відповідно до шкали ECTS. (рис. 3.5);



**Рис. 3.5 Кільцева діаграма секції “Студенти”**

5. Таблиця відображення успішності студентів навчальної групи за

значенням середнього балу (рис. 3.6).

Рейтинг студентів навчальної групи		
№	Ім'я	Середній Бал
1	Мечислава Прядун	82.85
2	Божена Ящук	82.47
3	Сідлецька Надія	81.76
4	Вередун Купава	80.03
5	Балицька Палажка	79.82
6	Роменець Юстина	79.56

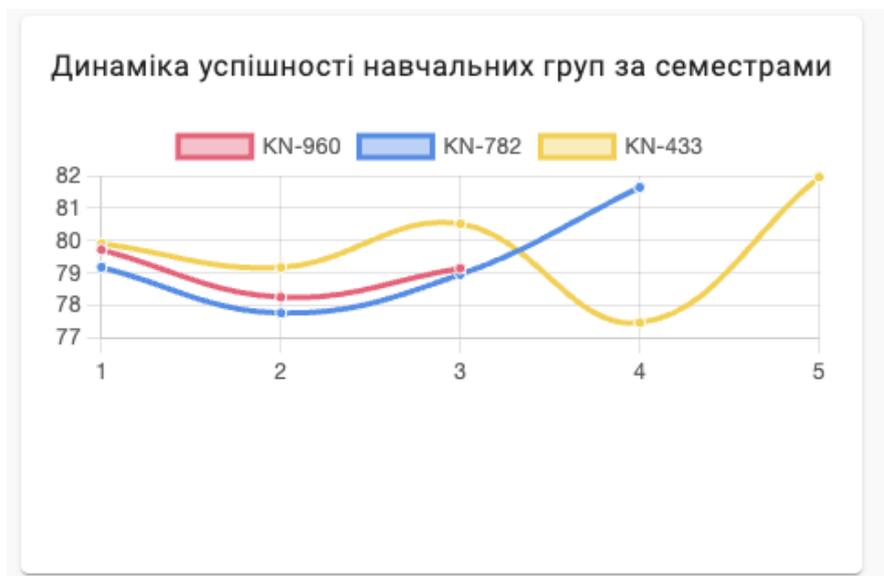
**Рис. 3.6 Таблиця рейтингу студентів навчальної групи секції  
“Студенти”**

Дана секція призначена для викладачів та ланок управління різного рівня структурної ієрархії підрозділів закладу вищої освіти. На підставі відомостей, які надають інформацію про загальний рівень знань, лідерів та аутсайдерів серед студентів дозволяють викладачу попередньо підготувати підходи і методи до навчання здобувачів вищої освіти з метою досягнення максимальної ефективності навчання під час вивчення дисципліни.

Секція “Дашборд” містить загальні відомості успішності кафедри закладу вищої освіти. Модуль системи складається з наступних компонентів:

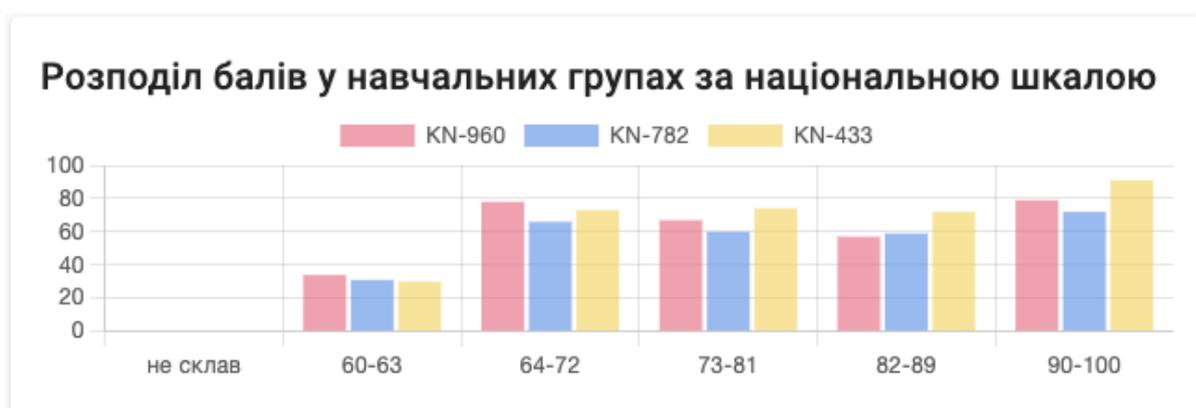
1. Статистичні показники – медіана, стандартне відхилення, дисперсія вибірки освітніх досягнень студентів за результатами екзаменаційної сесії. Призначення і способи використання даних показників під час проведення моніторингу успішності здобувачів освіти викладено у третьому пункті першого розділу кваліфікаційної роботи. Також, інформація про призначення і роль показників у здійсненні подальшого аналізу визначено у секції інформаційного меню “Документація”.
2. Кільцева діаграма розподілу рейтингових балів за шкалою ECTS.

3. Графік відстеження динаміки успішності усіх навчальних груп кафедри закладу вищої освіти (рис. 3.7). На даному графіку реалізовано флаги відображення груп. Користувач здійснивши клік по шифру навчальної групи визначає чи будуть відображатись відомості про конкретну навчальну групу на графіку.



**Рис. 3.7 Інформаційна панель візуалізації динаміка успішності навчальних груп секції “Дашборд”**

4. Стовпчаста діаграма розподілу освітніх досягнень студентів у групах за національною шкалою (рис. 3.8).



**Рис. 3.8 Діаграма розподілу освітніх досягнень за національною шкалою секції “Дашборд”**

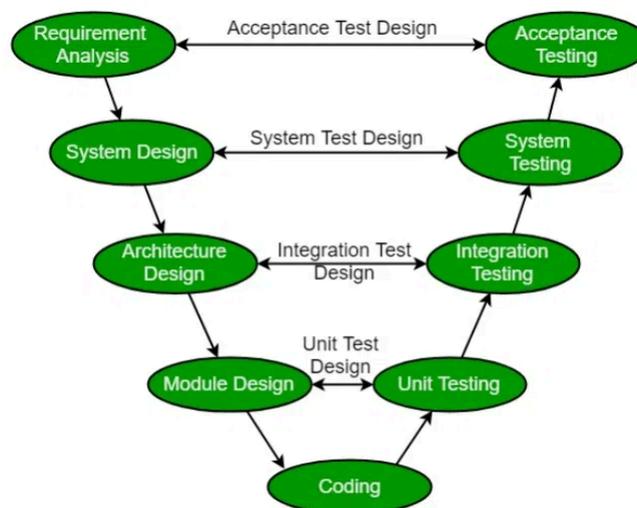
Використання секції “Дашборд” під час проведення аналізу результатів

моніторингу дозволяє визначати ефективність різних програм навчання робочих програм навчальних дисциплін та освітньо-професійних програм окремих спеціальностей. Також, якщо більшість навчальних груп демонструє низькі показники успішності у певному семестрі, це є підставою для аналізу дисциплін, які вивчаються у цьому семестрі.

Проведений аналіз дозволить визначити потенційну проблему, яка може полягати у одній з наступних причин: низька якість викладання, низький рівень зацікавленості студентів у вивченні дисципліни, невідповідність матеріалів курсу для ефективного засвоєння студентами.

### 3.3 Тестування застосунку

Під час створення інформаційної системи використовувалась V-подібна модель розробки програмного забезпечення. Вона є різновидом каскадної моделі де основною відмінністю від класичної моделі є те, що кожному етапу розробки відповідає окремий рівень тестування (рис. 3.9).



**Рис. 3.9 V-модель життєвого циклу програмного забезпечення**

*Джерело:* SDLC V-Model – Software Engineering. Режим доступу: –

<https://www.geeksforgeeks.org/software-engineering-sdlc-v-model/>

Здійснення переходів від одного етапу до наступного здійснюється лише

після завершення тестування попереднього етапу [33] . V-модель поділяється на наступні етапи:

1. Приймальне тестування – полягає у визначенні “критеріїв прийняття” правильності роботи інформаційної системи. На даному етапі аналізуються вимоги до системи, та функціональні особливості, які є обов’язковими до виконання.
2. Системне тестування – тестування працездатності системи на оточенні, яке відповідає релізному (платформи, сервери, операційні системи, сумісність версій програмного забезпечення). Даний вид тестування проводиться з метою забезпечення працездатності на обладнанні, яке було визначено замовником або розробниками проєкту.
3. Інтеграційне тестування – етап тестування, коли окремі модулі системи перевіряються у взаємодії з іншими. Розрізняють інтеграції з іншими системами а також сторонніми, партнерськими та власними бібліотеками. Один з найважливіших етапів тестування, оскільки перевіряється правильність взаємодії двох та більше компонентів системи. Дефекти, які виникають на даному етапі можуть бути приховані від “наочного виявлення” і призводять до неправильних обчислень та обробки даних.
4. Юніт тестування – етап тестування, під час якого здійснюється перевірка роботи окремих компонентів і методів програмного продукту. Ефективне проведення компонентного тестування дозволяє мінімізувати кількість дефектів на вищих етапах тестування.

Використовуючи принципи, які закладені у реалізацію програмного забезпечення за допомогою V-моделі було визначено наступні пріоритетні напрями тестування системи:

1. Коректне обчислення статистичних критеріїв.
2. Інтеграція Chart.js у інформаційну систему.
3. Достовірне відображення результатів успішності студентів на інформаційних панелях.

4. Підготовка та генерація тестових даних.
5. Компонентне тестування окремих методів і компонентів, які здійснюють обчислення і обробку даних.

Тестування програмного забезпечення необхідно здійснювати для того, щоб уникнути програмних помилок які можуть призвести до людських жертв та прийняття оманливих рішень. Відповідно до теми кваліфікаційної обчислення статистичних критеріїв є основою для проведення аналізу успішності студентів у закладі вищої освіти. Зважаючи на це, точність обчислення показників відіграє важливу роль у прийнятті управлінських рішень і потребує особливої уваги під час тестування .

Прикладом катастрофічних наслідків невірних обчислень слугує випадок використання ракет ППО “Patriot” у 1991 році. Тоді, через помилки обчислень установка не змогла перехопити ворожу ракету і потрапила у місце розташування особового складу. Як наслідок, невірні розрахунки у роботі програмного забезпечення призвели до загибелі багатьох військовослужбовців [34].

З метою мінімізації негативного впливу програмних обчислень під час розробки інформаційної системи було виконано додаткові перевірки:

1. Використання типу даних float, для числових змінних системи. Це дозволяє уникнути проблеми обрізання значень під час розрахунків через зведення чисел з одним типом точності до іншого [35].
2. Уникнення подвійних округлень числових значень під час виконання обчислень. Статистичні показники у інформаційній системі округлюються до сотих лише після виконання усіх проміжних розрахунків.

З метою дотримання принципу програмування DRY (англ. “don’t repeat yourself” - “не повторюйся”) [36], для здійснення обчислень було створено додатковий клас, який містить статичні методи для розрахунку значень статистичних критеріїв (див. Додаток Б).

Наступний етап тестування передбачав перевірку правильності візуалізації результатів успішності студентів на інформаційних панелях. Для побудови

графіків і діаграм на інформаційних панелях було обрано бібліотеку Chart.js. Chart.js – сучасна бібліотека побудови графіків і діаграм на вебсторінках [37]. Бібліотека надає API (англ. “application programming interface” - “програмний інтерфейс взаємодії”), виконання запитів до якого інтегрує елемент canvas на вебсторінку.

Використання графіків і діаграм у вебдодатках вимагає дотримання певних правил, які дозволяють досягти коректної візуалізації наборів даних:

1. Визначення конфігурації графіку (розміри, тип графіку чи діаграми, підписи осей координат та/або їх елементів).
2. Коректний формат набору даних для візуалізації.
3. Плагіни та анімації. Використання додаткових модулів з метою отримання додаткової динамічності графіку та анімації.

Приклад інтеграції діаграми Chart.js у компонент візуалізації представлено на рисунку 3.10.

```
export class DoughnutChartComponent implements OnInit {
  doughnutChartLabels: string[] = [];
  doughnutChartData: ChartDataset<'doughnut'>[] = [
    { data: [], label: 'Кількість оцінок' },
  ];
  //doughnutChartPlugins = [DataLabelsPlugin];
  doughnutChartOptions: ChartOptions<'doughnut'> = {
    responsive: true,
    maintainAspectRatio: false,
    plugins: {
      legend: {
        display: true,
        position: 'top',
      },
    },
  },
};
```

**Рис. 3.10** Конфігурація кільцевої діаграми у компоненті Angular

Важливим аспектом проектування інструментів призначених для закладів вищої освіти є дотримання етичних норм, принципів інклюзивного навчання[38]

і правил обробки персональних даних [39]. Після визначення необхідних користувальницьких даних виникає необхідність їх коректного відображення у застосунку.

Спираючись на те, що джерелом даних для візуалізації у системі виступають таблиці, то кожному студенту відповідає значна кількість записів, у яких містяться його результати успішності. З метою перевірки коректної візуалізації для тестування необхідно використовувати набори тестових даних які максимально наближені до реальних [40]. Враховуючи значний розмір таблиць, підготовка тестових даних шляхом заповнення усіх полів таблиць мануальним способом не є доцільною. Значні витрати часу і наявність людського фактору, становлять ризики для коректної підготовки даних.

Для автоматичної генерації тестових даних було розроблено скрипт з використанням бібліотеки Faker.js. Дана бібліотека призначена для генерації користувальницьких даних з метою використання у процесах тестування.

Таким чином, для генерації групи студентів створено відповідну функцію:

```
function generateGroupOfStudents(studentsInGroup) {  
  let students = [];  
  for (let id = 0; id < studentsInGroup; id++) {  
    let student = {};  
    let studentFullName = faker.person.fullName();  
    const userName = studentFullName.split(' ');  
    student.name = userName[0];  
    student.lastName = userName[1];  
    students.push(student);  
  }  
  return students;  
}
```

Дана функція приймає параметр “studentsInGroup” цілочисельного типу, який вказує на кількість студентів у групі, для якої генеруватимуться дані. Далі

використовується метод `faker.person.fullName()` для генерації строки, яка містить ім'я і прізвище людини. Отримана строка розділяється за допомогою методу `split()` на дві окремі сутності. Результатом виконання функції є масив об'єктів, які містять відомості про студентів групи.

Кінцева функція генерації тестових даних, які використовуються у здійсненні моніторингу результатів успішності студентів наведено на рисунку 3.11.

```
function generateSessionData({
  existedGroup = [],
  existedGroupName = 'KN-0',
  semester = 1,
}) {
  let result = [];
  let groupName = existedGroupName;
  let group = existedGroup;
  let control = getTypeOfControl();
  let teacher = faker.person.fullName();
  let date = `${faker.number.int({ min: 10, max: 31 })}.${faker.number.int({
    min: 1,
    max: 12,
  })}.${faker.helpers.arrayElement([2023, 2022])}`;
  let disciplineCode = faker.string.alpha({ length: 4, casing: 'upper' });
  for (let i = 0; i < group.length; i++) {
    let userData = {};
    userData.name = group[i].name;
    userData.lastName = group[i].lastName;
    userData.group = groupName;
    userData.semester = semester;
    userData.discipline = disciplineCode;
    userData.control = control;
    userData.mark = faker.number.int({ min: 60, max: 99 });
    userData.teacher = teacher;
    userData.date = date;
    result.push(userData);
  }
  return result;
}
```

**Рис. 3.11** Генерація результатів успішності студентів для здійснення тестування

Функція “GenerateSessionData” здійснює генерацію результатів успішності студентів за підсумками екзаменаційної сесії. За допомогою параметризації дана

функція може бути використана для генерації результатів успішності сесії для груп з різними: кількістю студентів, кількістю дисциплін у навчальному семестрі, типом підсумкового контролю. Передбачено, що у навчальній групі оцінювання студентів здійснює один викладач з визначеної дисципліни. Усі тестові дані записуються у файл з розширенням .xls (див. додаток А).

Для тестування коректної візуалізації і генерації достовірних рейтингових балів студентів з вивчення певної навчальної дисципліни було використано техніку и перевірки граничних значень. Сутність даної техніки, полягає у розбитті вхідних даних на еквівалентні класи, до яких існують визначені обмеження. Таким чином, якщо файл з даними містить рейтингові бали із від'ємними, не цілочисельними або більшими за 100 значеннями, то інформаційна система не здійснить опрацювання результатів і повідомить користувача, що файл з даними містить помилку.

Підсумовуючи викладений вище матеріал, доречно акцентувати увагу на тому, що процес розробки і тестування інформаційних систем призначених для здійснення моніторингу і візуалізації даних з метою проведення подальшого аналізу є складним і взаємопов'язаним процесом. Нехтування проведенням якісного тестування перед введенням застосунку в експлуатацію може призводити до незворотних наслідків.

## ВИСНОВКИ

Галузь освіти відіграє важливу роль у розвитку суспільства та відкриває перспективи для нових відкриттів і вдосконалення існуючих процесів діяльності соціуму. Розвиток освіти і науки є одним з завдань сталого розвитку України, що дозволить зміцнити роль держави на міжнародній арені та значно збільшити відсоток інвестицій у важливі галузі виробництва.

Сучасні Заклади вищої освіти мають складну структурну організацію з взаємопов'язаних елементів. Для провадження заходів спрямованих на вдосконалення і зростання якості освіти необхідно проводити комплекс заходів, що враховують: специфіку навчального закладу, поточний рівень якості навчання, проблемні сектори у провадженні навчальної та/або управлінської діяльності. Одним з етапів модернізації та вдосконалення рівня якості освіти Вищих навчальних закладів є моніторинг і аналіз існуючих результатів успішності студентів. Враховуючи перспективи розвитку навчальної аналітики і низьку кількість практичних наукових досліджень заданого напрямку, створення інформаційно-аналітичної системи для обробки і візуалізації результатів моніторингу успішності студентів стає дуже перспективним кроком, який дозволить збагатити арсенал інструментів навчальної аналітики.

У процесі виконання кваліфікаційної роботи було виконано усі поставлені завдання у повному обсязі, а саме: досліджено особливості і методики опрацювання результатів моніторингу освітніх досягнень здобувачів вищої освіти. Було вивчено існуючі способи та програмні засоби навчальної статистики призначених для опрацювання і візуалізації даних, проаналізувати інструменти побудови вебзастосунків, призначених для опрацювання і візуалізації даних. Практичним результатом дослідження є інформаційно-аналітична система для опрацювання та візуалізації результатів моніторингу успішності студентів.

У першому розділі кваліфікаційної роботи було досліджено особливості здійснення моніторингу у закладах вищої освіти та необхідність його проведення у закладах вищої освіти. Спираючись на дослідження сучасних науковців, визначено, що забезпечення високої якості освіти навчального закладу є складним і багатокomпонентним процесом, на який впливають рівень не лише матеріально-технічного забезпечення, методи і способи навчання, рівень майстерності науково-педагогічних працівників, але й сила зв'язків взаємодії між студентами і викладачами. Здійснення постійного моніторингу результатів успішності студентів надає вичерпну інформацію про поточний стан здійснення освітнього процесу у підрозділах вищих навчальних закладів. Здійснення контролю за рівнем успішності студентів на різних рівнях надає вичерпну інформацію для здійснення управлінської діяльності. Визначено, які статистичні показники корелюють з якістю навчання і яким чином можуть бути використані під час проведення аналізу успішності навчального закладу. Використання компонентів візуалізації є необхідним компонентом сучасної інформаційно-аналітичної системи призначеної для закладу освіти. Це пов'язано із значною кількістю даних, необхідних для опрацювання.

Правильно побудована система візуалізації може стати потужним програмним засобом, який дозволяє виявляти ознаки, які можуть свідчити про проблемні ділянки освітнього процесу, наявність академічної недоброчесності і корупції. Аналіз результатів візуалізації на предмет наявності корупційних дій або елементів академічної недоброчесності вимагає додаткового дослідження і проведення експертних комісій керівництвом відповідних підрозділів навчального закладу.

У другому розділі проаналізовано існуючі системи візуалізації статистичних даних, таких як “Looker Studio”, “Click Up”, “Data Box”. Розглянуто сучасні фреймворки і бібліотеки призначені для побудови вебзастосунків. Визначено, що фреймворк Angular є багатофункціональним інструментом для побудови односторінкових вебдодатків. Використання

двостороннього зв'язування компонентів за допомогою директив дозволяє досягти бажаних результатів візуалізації і динамічності застосунку.

У третьому розділі визначено вимоги до інформаційно-аналітичної системи, яка є практичним результатом кваліфікаційної роботи. Розкрито особливості і виклики практичної реалізації застосунку, продемонстровано наявні компоненти інтерфейсу і панелі візуалізації результатів успішності студентів відповідно до функціональних вимог. Продемонстровано важливість і етапи проведення тестування системи впродовж усього етапу створення застосунку. Обґрунтовано використання технік тест-дизайну для перевірки необхідної функціональності, підходів та інструментів необхідних для генерації тестових даних.

Розроблена інформаційно-аналітична система для опрацювання та візуалізації результатів моніторингу успішності студентів є кінцевим результатом виконання усіх поставлених завдань визначених кваліфікаційною роботою. Застосунок може використовуватись у закладах вищої освіти, як допоміжний інструмент здійснення локального моніторингу, який спрямований на виявлення проблемних ділянок та вдосконалення освітнього процесу. Моніторинг результатів успішності студентів дозволяє наочно демонструвати вплив управлінських рішень, ефективність використання різних методик навчання, які мають безпосередній вплив на забезпечення якості освіти.

Застосунок має широкі перспективи подальшого розвитку, який полягає в здатності інтеграції до інформаційних систем, які використовують заклади вищої освіти. Реалізація розмежування прав доступу до перегляду результатів успішності для категорії викладачів і студентів дозволить розширити взаємодію між ними, а використання індивідуальних досягнень нададуть студентам додаткову мотивацію до навчання, оскільки вони зможуть спостерігати динаміку своєї успішності впродовж усього періоду здобуття вищої освіти.

**ДОДАТКИ**  
*Додаток А*  
**Зразок джерела даних**

name	last_name	group	semester	discipline	type_of_contro	mark	teacher	date
Степанець	Адріана	KN-960	1	KKPF	залік		92 Щастислав Горб	23.1.2022
Верещук	Агапія	KN-960	1	KKPF	залік		70 Щастислав Горб	23.1.2022
Адам	Поривайло	KN-960	1	KKPF	залік		67 Щастислав Горб	23.1.2022
Мазило	Афанасій	KN-960	1	KKPF	залік		67 Щастислав Горб	23.1.2022
Лада	Гарасимівна	KN-960	1	KKPF	залік		98 Щастислав Горб	23.1.2022
Гойко	Поліна	KN-960	1	KKPF	залік		91 Щастислав Горб	23.1.2022
Мілана	Марківна	KN-960	1	KKPF	залік		89 Щастислав Горб	23.1.2022
Аліна	Мінняйло	KN-960	1	KKPF	залік		75 Щастислав Горб	23.1.2022
Трублаєвська	Калина	KN-960	1	KKPF	залік		75 Щастислав Горб	23.1.2022
Арсен	Гайовий	KN-960	1	KKPF	залік		70 Щастислав Горб	23.1.2022
Іванна	Гриневська	KN-960	1	KKPF	залік		73 Щастислав Горб	23.1.2022
Поривайло	Аполлінарія	KN-960	1	KKPF	залік		86 Щастислав Горб	23.1.2022
Галаціон	Дарислава	KN-960	1	KKPF	залік		73 Щастислав Горб	23.1.2022
Білослав	Савицький	KN-960	1	KKPF	залік		62 Щастислав Горб	23.1.2022
Ніна	Павлишина	KN-960	1	KKPF	залік		90 Щастислав Горб	23.1.2022
Степанець	Адріана	KN-960	1	HNNQ	залік		81 Захарій Несторс	29.2.2022
Верещук	Агапія	KN-960	1	HNNQ	залік		94 Захарій Несторс	29.2.2022
Адам	Поривайло	KN-960	1	HNNQ	залік		80 Захарій Несторс	29.2.2022
Мазило	Афанасій	KN-960	1	HNNQ	залік		86 Захарій Несторс	29.2.2022
Лада	Гарасимівна	KN-960	1	HNNQ	залік		60 Захарій Несторс	29.2.2022
Гойко	Поліна	KN-960	1	HNNQ	залік		86 Захарій Несторс	29.2.2022
Мілана	Марківна	KN-960	1	HNNQ	залік		65 Захарій Несторс	29.2.2022
Аліна	Мінняйло	KN-960	1	HNNQ	залік		99 Захарій Несторс	29.2.2022
Трублаєвська	Калина	KN-960	1	HNNQ	залік		66 Захарій Несторс	29.2.2022
Арсен	Гайовий	KN-960	1	HNNQ	залік		61 Захарій Несторс	29.2.2022
Іванна	Гриневська	KN-960	1	HNNQ	залік		95 Захарій Несторс	29.2.2022
Поривайло	Аполлінарія	KN-960	1	HNNQ	залік		65 Захарій Несторс	29.2.2022
Галаціон	Дарислава	KN-960	1	HNNQ	залік		74 Захарій Несторс	29.2.2022
Білослав	Савицький	KN-960	1	HNNQ	залік		81 Захарій Несторс	29.2.2022
Ніна	Павлишина	KN-960	1	HNNQ	залік		80 Захарій Несторс	29.2.2022
Степанець	Адріана	KN-960	1	WUQA	екзамен		81 Марія Добросла	10.10.2022
Верещук	Агапія	KN-960	1	WUQA	екзамен		78 Марія Добросла	10.10.2022
Адам	Поривайло	KN-960	1	WUQA	екзамен		78 Марія Добросла	10.10.2022
Мазило	Афанасій	KN-960	1	WUQA	екзамен		67 Марія Добросла	10.10.2022
Лада	Гарасимівна	KN-960	1	WUQA	екзамен		99 Марія Добросла	10.10.2022
Гойко	Поліна	KN-960	1	WUQA	екзамен		70 Марія Добросла	10.10.2022
Мілана	Марківна	KN-960	1	WUQA	екзамен		63 Марія Добросла	10.10.2022

## Додаток Б

## Лістинг обчислення показників

```

export class GradeCalculation{
  constructor() {
  }
  //Calculations for statistics criteria
  static calculateMode(data: number[]): number {
    const frequencyMap: { [key: number]: number } = {};
    data.forEach((value) => {
      frequencyMap[value] = (frequencyMap[value] || 0) + 1;
    });
    let mode: number | null = null;
    let maxFrequency = 0;

    for (const value in frequencyMap) {
      if (frequencyMap[value] > maxFrequency) {
        mode = +value;
        maxFrequency = frequencyMap[value];
      }
    }
    return mode || 0;
  }
  static calculateMedian(data: number[]): number {
    const sortedData = data.slice().sort((a, b) => a - b);
    const length = sortedData.length;
    if (length % 2 === 0) {
      const midIndex = length / 2;
      return (sortedData[midIndex - 1] + sortedData[midIndex]) / 2;
    } else {
      const midIndex = Math.floor(length / 2);
      return sortedData[midIndex];
    }
  }
  static calculateAverage(grades: number[]): number{
    const sum = grades.reduce((accumulator, currentGrade) => accumulator +
currentGrade, 0);
    return parseFloat((sum / grades.length).toFixed(2));
  }
  static calculateVariance(data: number[]): number {
    const n = data.length;
    const mean = data.reduce((sum, value) => sum + value, 0) / n;
    const sumSquaredDeviations =
data.reduce((sum, value) => sum + Math.pow(value - mean, 2), 0);

```

### Продовження додатку Б

```

return parseFloat((sumSquaredDeviations / n).toFixed(2));
}
static calculateStandardDeviation(grades: number[]): number {
  if (grades.length < 2) {
    return 0;
  }
  const mean:number = this.calculateAverage(grades);
  const squaredDifferences = grades.map(grade => Math.pow(grade - mean, 2));
  const sumSquaredDifferences = squaredDifferences.reduce(
    (accumulator, squaredDifference) => accumulator + squaredDifference, 0);
  const variance = sumSquaredDifferences / (grades.length - 1);
  return parseFloat(Math.sqrt(variance).toFixed(2)) ;
}
static getCalculatedGroupAverageMarks(students: IStudent[]): IGroupSummary[] {
  const groupStatsMap: { [group: string]: { totalMark: number; count: number } } =
{};
  students.forEach((student) => {
    let { group, mark } = student;
    if (!groupStatsMap[group]) {
      groupStatsMap[group] = { totalMark: 0, count: 0 };
    }
    groupStatsMap[group].totalMark += mark;
    groupStatsMap[group].count++;
  });
  const groupAverageMarks: IGroupSummary[] = [];
  for (const group in groupStatsMap) {
    if (groupStatsMap.hasOwnProperty(group)) {
      const { totalMark, count } = groupStatsMap[group];
      const averageMark = +(totalMark / count).toFixed(1);
      const studentCount = this.getCountStudentsPerGroup(students)[group];
      groupAverageMarks.push({ group, averageMark, studentCount });
    }
  }
  return groupAverageMarks;
}
}

```

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. A Current Overview of the Use of Learning Analytics Dashboards. URL: <https://www.mdpi.com/2227-7102/14/1/82>
2. Косінов М., Важливість розробки інформаційно-аналітичної системи для опрацювання та візуалізації результатів моніторингу успішності студентів. Вісник Кам'янець-Подільського національного університету імені Івана Огієнка. Фізико-математичні науки. Випуск 16. Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2023, С. 42-47.
3. Косінов М., Проектування інформаційно-аналітичної системи для опрацювання та візуалізації результатів моніторингу успішності студентів. Збірник матеріалів наукової конференції за підсумками науково-дослідної роботи здобувачів вищої освіти фізико-математичного факультету Кам'янець Подільського національного університету імені Івана Огієнка у 2023-2024 н.р., 9-10 квітня 2024 року [Електронний ресурс]. Кам'янець-Подільський Кам'янець-Подільський національний університет імені Івана Огієнка, фізико-математичний факультет, 2024. с. 57-59 URL: <http://elar.kpnu.edu.ua/xmlui/handle/123456789/8094>
4. Косінов М., Смалько О. Розробка інформаційних панелей для візуалізації результатів успішності навчання в освітніх інформаційно-аналітичних системах. Актуальні аспекти розвитку STEAM-освіти в умовах євроінтеграції: збірник матеріалів II Міжнародної науково-практичної інтернет-конференції (м. Кропивницький, 26 квітня 2024 року). Кропивницький : ДонДУВС, 2024., С. 258-260.
5. Ольга Осередчук, Модель моніторингу якості вищої освіти в Україні, DOI: 10.33989/2075-146x.2022.29.264320. URL: <https://shorturl.at/nwGOx>

6. Оцінювання втрат та потреб української освіти на основі RDNA3.  
URL: <https://shorturl.at/HVLPb>
7. Стратегія розвитку вищої освіти в Україні на 2021–2031 роки.  
URL: <http://surl.li/dgylw>
8. Державна служба якості освіти України. URL: <https://11l.ink/qxt9f>
9. Зінченко В.О. Моніторинг якості навчального процесу у вищому навчальному закладі : монографія. Луганськ : Вид-во ДЗ “ЛНУ імені Тараса Шевченка”, 2013. 360 с.
10. Моніторинг навчальної діяльності: навчальний посібник. Д.М. Бодненко, О.Б. Жильцов, О.Л. Лещинський, Н.П. Мазур. К: Київський університет імені Бориса Грінченка, 2014. 276 с. ISBN 978-966-7548-98-8
11. Про затвердження Порядку проведення моніторингу якості освіти : Наказ МОН України від 16 січня 2020 р. № 54. Київ. URL: <https://zakon.rada.gov.ua/laws/show/z0154-20#n14>
12. Приходько В.М. Парадигми моніторингу якості освіти і педагогічного процесу: (У запитаннях і відповідях). Науково-методичний посібник. Запоріжжя, 2010. 215 с.
13. Освіта в умовах пандемії у 2020/2021 році: аналіз проблем і наслідків URL: <https://t.ly/KAyaO>
14. Диджиталізація освітнього процесу і дистанційне навчання в Україні: виклики, проблеми, перспективи. URL: <http://mir.dspu.edu.ua/article/view/271161/268264>
15. І. П. Анненкова КРИТЕРІЇ І ПОКАЗНИКИ ЯКОСТІ ОСВІТИ У ВНЗ. ОНУ імені І. І. Мечникова, м. Одеса. URL: <https://shorturl.at/cLrLW>
16. Якість вищої освіти: теорія і практика: навчально-методичний посібник / за наук. ред. А. Василюк, М. Дей; кол. авторів: А. Василюк, М. Дей, В. Базелюк (та ін.); НАПН України, Університет менеджменту освіти. Київ; Ніжин: Видавець ПП Лисенко М. М.,

2019. 176 с.

17. М. М. Ручкіна Навчальний посібник з дисципліни «Основи педагогічних вимірювань та моніторинг якості освіти» для здобувачів освіти першого (бакалаврського) рівня спеціальності 014 Середня освіта (Мова і література (турецька, китайська, корейська, англійська)). Одеса, 2023. 161 с.
18. Проценко І. І., Гудименко К.М. Моніторинг як інструмент визначення якості освіти. Педагогічні науки: теорія, історія, інноваційні технології, 2015, № 4 (48). URL: <https://t.ly/GYBp8>
19. Ольга Осередчук Сутність і структура понять “якість”, “якість освіти”, “моніторинг”, “моніторинг в освіті”, “моніторинг якості вищої освіти” УДК 378. 112 (477) URL: <http://surl.li/uedqd>
20. Пасічник Н. Моніторинг якості освіти як комплексне оцінювання освітнього процесу. URL: <http://surl.li/uebzu>
21. What is Google Looker Studio URL: <https://11l.innk/SX6Ow>
22. Looker Studio Docs URL: <https://cloud.google.com/looker/docs>
23. DataBox dashboard examples URL: <https://databox.com/dashboard-examples>
24. ClickUp documentation URL: <https://clickup.com/features/docs>
25. ClickUp data visualization features. URL: <https://clickup.com/on-demand-demo>
26. The seven phases of the software development life cycle. URL: <https://shorturl.at/gFGn6>
27. ECMAScript 2015 Language Specification. URL: <https://262.ecma-international.org/6.0/>
28. The Pros and Cons of Using React for Front-End Development. URL: <https://www.devstringx.com/pros-and-cons-of-using-react>
29. Pros and Cons of Vue.js Framework Programming. URL: <https://t.ly/UmiVu>

30. Angular Single Page Applications (SPA): What are the Benefits? URL: <https://t.ly/wpBAB>
31. Levels and Types of Requirements. URL: <http://surl.li/uecys>
32. Introduction to Node.js. URL: <http://surl.li/uedsq>
33. SDLC V-Model - Software Engineering. URL: <http://surl.li/uedbz>
34. Patriot Missile Defense: Software Problem Led to System Failure at Dhahran, Saudi Arabia: IMTEC-92-26 / A report of the General Accounting office, GAO/IMTEC-92-261992, 1992. URL: <https://www.gao.gov/assets/220/215614.pdf>
35. Floating Point errors in JavaScript/Node.js. URL: <http://surl.li/uedck>
36. DRY Software Design Principle. URL: <https://shorturl.at/Qpq2g>
37. Chart.js Documentation. URL: <https://www.chartjs.org/>
38. Засади інклюзивної освіти в Україні. URL: <https://shorturl.at/9SPGW>
39. ЗАКОН УКРАЇНИ №2297–VI від 01.06.2010 р., Про захист персональних даних. URL: <http://surl.li/uedqt>
40. What is Test Data Generation? URL: <http://surl.li/uedqm>