

[0000-0002-2180-0969] **Р. Л. Пташкін<sup>1</sup>**,

e-mail: ndekc.ck@gmail.com

[0000-0003-4043-5300] **А. В. Гончаров<sup>2</sup>**, канд. техн. наук, доцент,

e-mail: a.honcharov@chdtu.edu.ua

[0000-0003-4621-9510] **О. С. Гавриш<sup>2</sup>**, канд. фіз.-мат. наук, доцент

e-mail: o.havrysh@chdtu.edu.ua

<sup>1</sup>Черкаський науково-дослідний експертно-криміналістичний центр МВС України  
вул. Пастерівська, 104, м. Черкаси, Україна

<sup>2</sup>Черкаський державний технологічний університет  
б-р Шевченка, 460, м. Черкаси, 18006, Україна

## ІНФОРМАЦІЙНО-АНАЛІТИЧНА СИСТЕМА ОБЛІКУ ЕКСПЕРТНИХ ПРОВАДЖЕНЬ У ПІДРОЗДІЛАХ ЕКСПЕРТНОЇ СЛУЖБИ МВС УКРАЇНИ

*Робота присвячена проблемам побудови моделей, розробки, реалізації та впровадження інформаційно-аналітичної системи обліку експертних проваджень у підрозділах Експертної служби МВС України. Аналізуючи сучасні вимоги та принципи розробки програмних засобів для підрозділів МВС України, авторами було сформовано перелік вимог та базових принципів стосовно розробки необхідного програмного забезпечення. Окрім того, було здійснено аналіз сучасних технологій, які можливо використати для розробки системи. Відокремлено ті технологічні рішення, що повністю задовольняють поставлені вимоги. Відтак авторами для розробки програмного засобу обрано архітектуру на кшталт web-додатка, для роботи з яким користувачеві необхідний лише web-браузер, в якому буде відображений графічний інтерфейс користувача, а вся функціональність системи забезпечується web-сервером.*

*При розробці авторами приділено особливу увагу питанню захисту інформації, оскільки воно продиктовано ключовими вимогами до програмного забезпечення, що може бути використане в підрозділах МВС України. Відтак розроблена система фундаментально базується на моделі безпеки «Zero Trust» і здійснює багаторівневе фільтрування та перевірку будь-яких даних, що обробляються в системі. Також система має алгоритми автоматизованої реакції на спроби несанкціонованого вторгнення.*

*Як ключові технології було обрано систему керування базами даних PostgreSQL для збереження та маніпулювання даними, web-сервер nginx для забезпечення функціонування сервера, мову програмування PHP для написання коду серверної частини системи та мови HTML, CSS, JavaScript для реалізації графічного інтерфейсу користувача. Виходячи з вимог безпеки, при розробці не використовувалися сторонні бібліотеки чи фреймворки. Безпосередньо для реалізації серверної частини додатка авторами обрано децю модифіковану модель MVC та реалізовано принцип абсолютної модульності системи.*

*Інформаційно-аналітичну систему було успішно впроваджено в діяльність усіх підрозділів Експертної служби МВС України. Наразі система активно використовується для автоматизованого створення звітності й формування статистичних та аналітичних даних про окремі напрями діяльності служби.*

**Ключові слова:** інформаційно-аналітична система, аналіз даних, статистика, аналітика, розробка програмного забезпечення, бази даних, web, php, PostgreSQL.

**Вступ.** Сучасний стан розвитку та функціонування структурних одиниць Міністерства внутрішніх справ України (МВС України) в цілому й Експертної служби МВС України зокрема тією чи іншою мірою перебуває в активній фазі цифрової трансформації. Поступово в підрозділах МВС України відбуваються процеси щодо затвердження і запрова-

дження інформаційних систем [1], банків даних, пошукових систем чи систем обліку [2]. Синхронно з процесами інформатизації повсякденної діяльності державних органів здійснюється й сукупність заходів щодо захисту інформації та систем її обробки [3].

Фактично інформаційні технології посіли чільне місце в усіх сферах діяльності орга-

нів та підрозділів Експертної служби МВС України. А з огляду на посилені тенденції реформування державних структур України, без активного застосування інформаційних технологій в експертній діяльності нині не обійтися.

Завдяки правильному розумінню пріоритетних напрямів розвитку інформатизації діяльності фактично маємо можливість підняти технологічний рівень обігу інформації в державі загалом і в Експертній службі МВС України зокрема на належний рівень. Адже саме застосування інформаційно-аналітичних систем надає унікальну можливість здійснювати активний моніторинг перебігу робочих процесів та своєчасні корегувальні дії.

На сьогодні практично неможливо своєчасно та правильно здійснювати будь-які адміністративні дії, не маючи доступу до різнопланової, актуальної та достовірної інформації, збір і надання якої ґрунтуються на сучасних спеціалізованих базах даних із використанням відповідного програмного забезпечення.

Аналізуючи останні наукові напрацювання, можна стверджувати, що питання розробки спеціалізованого програмного забезпечення, яке б своїми функціональними можливостями давало змогу здійснювати облік та опрацювання даних, що з'являються в процесі функціонування підприємства, вже багаторазово опрацьовані в різних напрямках. Так, науковцями вже було опрацьоване питання розробки інформаційно-аналітичної системи для закладів вищої освіти [4], а в роботі [5] загалом розглянуто питання доцільності використання інформаційних та інформаційно-аналітичних систем в управлінні сучасним підприємством.

Також авторами проаналізовано численні напрацювання таких вітчизняних науковців, як Д. В. Дабіжа, С. В. Пеньков, В. М. Рудік, В. А. Кудінов, Я. Ф. Кулешик, О. І. Зачек, В. Б. Вишня, які в своїх працях тим чи іншим чином розкривають питання особливостей використання інформаційних, інформаційно-аналітичних та пошукових систем у підрозділах органів внутрішніх справ України.

Окрім того, опрацьовано роботи Мартіна Клеппмана [6], Дерак ДеДжонге [7], Лейне Кампбелл та Чаріті Мейорс [8], Джоел Грус [9], які присвячені особливостям розробки та підтримки високонавантажених систем, прийо-

мам опрацювання великих масивів даних та створенню надійних засобів обробки інформації.

Саме аналіз вже існуючих напрацювань дав можливість виявити основну проблематику розробки банків даних та систем обробки інформації як у підрозділах МВС України загалом, так і в Експертній службі МВС України зокрема.

Проблематика полягає в тому, що Експертна служба є установою, діяльність якої безпосередньо пов'язана з наданням послуг, що мають велике значення для функціонування суспільства та безпеки населення. Порушення функціонування служби може справити негативний вплив на стан національної безпеки і оборони України. Тобто підрозділи Експертної служби є об'єктами критичної інфраструктури.

Окрім того, дані, що опрацьовуються в підрозділах Експертної служби МВС України, в більшості випадків мають або відомості стосовно персональних даних громадян, або ж інформацію з обмеженим доступом. Витік такої інформації може справити негативний вплив на життєво важливі інтереси людини і громадянина, суспільства та держави в цілому.

Саме з огляду на вищезазначене, до програмних засобів, що використовуються в Експертній службі МВС України для зберігання та опрацювання інформації, висуваються особливі вимоги – програмне забезпечення повинно мати як алгоритми криптографічного захисту інформації, так і систему захисту від витоку інформації. Окрім того, такі програмні засоби не повинні містити в своїй структурі елементів, що розробляються сторонніми організаціями без дотримання таких вимог.

**Мета дослідження:** побудова моделей інформаційно-аналітичної системи обліку експертних проваджень (судових експертиз), а також її проєктування, розробка та впровадження в Експертній службі МВС України.

Розроблюване програмне забезпечення повинно мати такі функціональні можливості:

- внесення та структуроване збереження відомостей про експертні провадження;
- проведення пошуку та сортування експертних проваджень за будь-якою комбінацією пошукових параметрів;
- формування звітності з кількісними показниками в реальному часі чи на момент часу в минулому;

- формування аналітичних даних у реальному часі чи на момент часу в минулому;
- здійснення порівняння поточних кількісних чи аналітичних відомостей з певним моментом чи періодом у минулому;
- розділення прав та режимів доступу користувачів;
- наявність механізмів захисту облікових записів користувачів та користувацьких даних;
- одночасне використання в усіх підрозділах Експертної служби МВС України.

Окрім того, функціональні можливості програмного засобу не повинні залежати від клієнтського обладнання чи програмного забезпечення. Тобто система має бути повністю кросплатформеною.

**Виклад основного матеріалу.** Після формування основних вимог до програмного забезпечення процес його розробки умовно можна розділити на декілька етапів:

- вибір технологій розробки;
- проектування бази даних;
- моделювання структури додатка;
- безпосередньо розробка.

Формуючи *вимоги до технологій розробки*, враховуючи специфіку майбутнього додатка, варто звернути особливу увагу на відкриті технології (Open Source). Вихідний код таких програм доступний для перегляду та вивчення, що дає можливість переконатися у відсутності вразливостей і неприйнятних функцій (наприклад прихованого спостереження чи збору даних).

З огляду на вимогу кросплатформеності, найефективнішим буде застосування web-технологій для проектування додатка.

Web-підхід фактично розділяє програмне забезпечення на дві функціонально різні частини – frontend (клієнтська сторона додатка, графічний інтерфейс) і backend (програмно-апаратна частина). Такий розділ дає змогу знизити системну складність, підвищити надійність і адаптивність (гнучкість) кінцевої програми.

Тобто функціонування системи буде забезпечуватися web-сервером, для роботи з додатком клієнтові необхідним буде лише веб-браузер, в якому буде здійснюватися відображення графічного інтерфейсу (frontend). В свою чергу, web-сервер буде отримувати запити від клієнта та передаватиме їх безпосередньо програмному засобу (backend) для подальшого опрацювання. Програмний засіб

може бути реалізований тією чи іншою мовою програмування й фактично не залежить від типу web-сервера. В обов'язки backend входить лише опрацювання запиту, за необхідністю – взаємодія з базою даних і, як результат, – формування відповіді (рисунок 1).

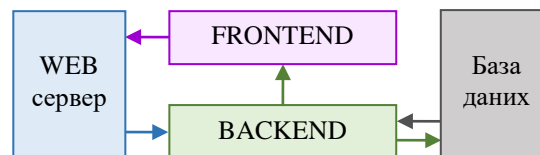


Рисунок 1 – Структура системи

Як web-сервер застосуємо *nginx* – програмний засіб з відкритим вихідним кодом, основна функціональність якого дає змогу реалізувати швидкий та надійний HTTP-сервер [7].

Для реалізації безпосередньо програмного засобу (backend) використаємо *PHP* – скриптову мову програмування, що була створена для генерації даних (frontend) на стороні web-сервера [18]. Однією з переваг мови PHP є те, що вона є скриптовою мовою, яка інтерпретується і транслюється по ходу виконання, це значно спрощує написання коду та процес його налагодження. Також зауважимо, що PHP є об'єктно-орієнтованою мовою, яка має серед свого арсеналу досить потужні можливості з контролю доступу до даних та методів їх обробки.

Як базу даних додатка використаємо *PostgreSQL* – об'єктно-реляційну систему керування базами даних (СКБД), яка є однією з найбільш передових систем з відкритим вихідним кодом [14]. Серед стандартних можливостей СКБД PostgreSQL наявні методи створення функцій, індексів, тригерів, потужні аналітичні засоби (наприклад віконні та агрегатні функції), що повною мірою задовольняє потреби, які можуть виникнути в процесі проектування та реалізації бази даних додатка.

Загалом на цьому етапі вже сформована архітектура додатка, яка визначається принципами взаємодії між різними ключовими технологіями розробки та рішеннями щодо організації програмної системи.

*Проектування бази даних* варто починати з детального вивчення її майбутніх даних, адже саме вони є тією основою, заради якої створюється майбутній програмний засіб. Першочергово необхідно провести сегментацію даних, їх структурування, виявлення

ключових ланок та формування схем зв'язків одних даних з іншими. Схеми зв'язків повинні забезпечувати цілісність та узгодженість інформації, мати елементи її ідентифікації.

Як перший рівень сегментації доцільним буде здійснити розподіл усіх даних на дві підмножини – абсолютні дані та описові дані.

*Описові дані* – це певний клас відомостей, що притаманний абсолютним даним. Наприклад, для експертного провадження описовими даними будуть відомості про напрям дослідження чи його дослідження, судового експерта тощо. Тобто описовими є дані, які можна винести в окремі довідники й для опису абсолютних даних використовувати лише ідентифікатор запису в цих довідниках.

*Абсолютними даними* будемо вважати такі відомості, що є унікальними для кожного запису. Наприклад, для експертного провадження абсолютним буде вхідний та вихідний номери, номер кримінального провадження тощо.

Зв'язок між абсолютними та описовими даними варто реалізовувати за допомогою таблиць зв'язків та зовнішніх ключів (foreign key) [15]. Такий підхід дасть змогу без зайвих зусиль присвоювати кожному елементу абсолютних даних по декілька елементів описових даних за принципом «багато – до – багатьох». Таблиця зв'язків буде містити лише два ідентифікатори – ідентифікатор абсолютного запису та ідентифікатор описового запису (рисунок 2). Пара цих ідентифікаторів буде унікальною й слугуватиме первинним ключем таблиці зв'язків.

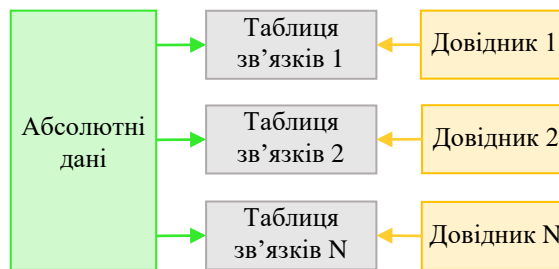


Рисунок 2 – Метод організації даних

Однак метод організації зв'язку між абсолютними та описовими даними з використанням таблиць зв'язків не завжди є доцільним, оскільки в деяких випадках абсолютні дані не можуть мати декілька елементів описових даних одного типу (наприклад, дослідження може мати лише один регіон реєстрації чи тільки один вид складності). В такому випадку доцільно зв'язувати описові та абсолютні дані напряму, без використання таблиць зв'язків, за принципом «один – до – багатьох».

Надалі сегментація полягає в розділенні описових даних за їх видами та організації необхідних логічних зв'язків з абсолютними даними.

Поєднуючи зазначені вище методи організації даних та зв'язків між ними, сформуємо структуру бази даних, яка міститиме інформацію про експертні провадження та їх атрибути, судових експертів та їх штатний розподіл (рисунок 3).

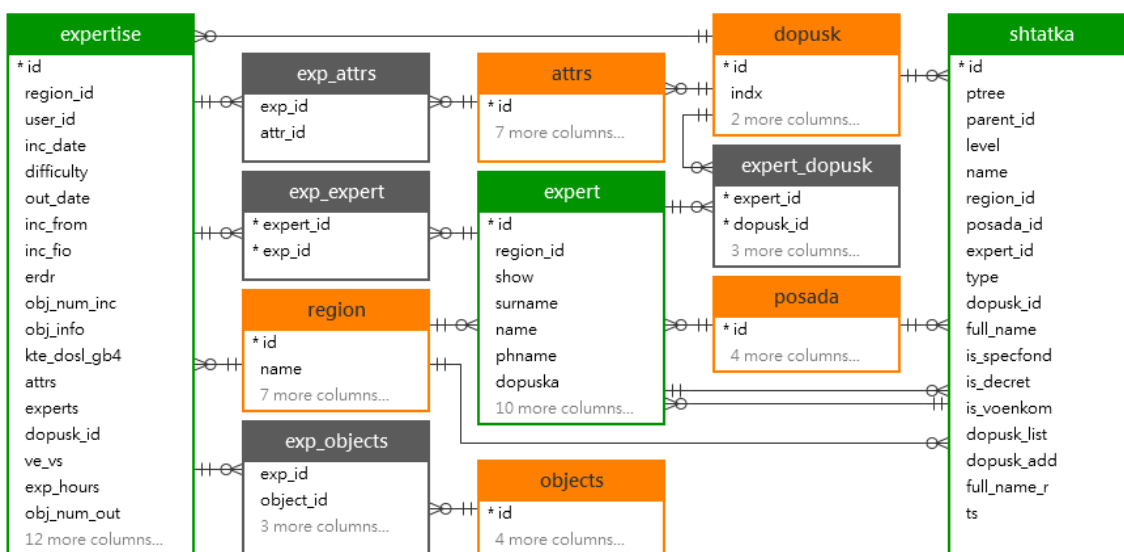


Рисунок 3 – Структура частини бази даних

Реалізована структура буде нормалізованою, цілісною, узгодженою, атомарною та логічною. Фактично така структура гарантує, що дані будуть перебувати в узгодженому, несуперечливому стані до початку будь-якої транзакції (модифікації даних) та після її завершення.

Принцип узгодженості гарантує інваріантність бази даних – будь-які дані, внесені в базу, повинні будуть задовольняти всі визначені правила, враховуючи обмеження, каскади, тригери та будь-яку їхню комбінацію.

В свою чергу, зв'язок унікального ключа з зовнішнім фактично унеможлиблює «засмічення» бази даних та забезпечує її автоматичну актуалізацію.

В ході реалізації програмного засобу на рівень СКБД необхідно виносити всі можливі алгоритми модифікації даних – формування статистичних відомостей чи допоміжної інформації найдоцільніше реалізовувати за допомогою тригерів, оскільки вони гарантують безвідмовне виконання тих чи інших алгоритмів при модифікації даних в таблиці. Окрім того, саме при реалізації структури таблиць бази даних варто реалізовувати не тільки структуру зовнішніх ключів, а й перевірку значення даних (*constraint ... check*). Хоча типи даних самі по собі обмежують значення даних, що зберігатимуться в таблиці, однак у більшості випадків такі обмеження занадто грубі.

*Моделювання структури додатка* варто почати з вибору шаблону (патерну) розробки. Основна користь від використання шаблонів полягає в зниженні складності розробки за рахунок стандартизованих абстракцій для вирішення цілого класу проблем. Фактично шаблон програмування визначає методи та правила організації зв'язку між елементами програми без формування вимог до їх реалізації, тобто формує архітектуру системи.

Вивчаючи роботи Е. Гамми [17], С. Пітта [18] та М. Зандстри [19], було розглянуто базові архітектурні шаблони, які можливо використати при створенні web-дodatка, а саме такі, як «MVC», «MVP», «MVVM» та «PAC». Також було розглянуто модифікації вищезазначених шаблонів – адаптації з варіативними контролерами та розподіленими моделями.

Кожен із патернів програмування має свої переваги та недоліки. Фактично за рахунок шаблонів проводиться уніфікація реалізації модулів, елементів проєкту, що знижує кі-

лькість помилок. Застосування шаблонів концептуально схоже на використання готових бібліотек коду – маючи достатній набір бібліотек, процес створення додатка зводиться до формування необхідного алгоритму з доступних методів.

З другого ж боку, використання шаблонного рішення значно ускладнює проєктування та майбутнє розуміння програми в разі виходу за межі шаблону. Рано чи пізно додаток еволюціонує до такого стану, коли необхідна реалізація таких зв'язків, що не передбачені базовою архітектурою.

На ранньому етапі моделювання необхідно підібрати відповідні об'єкти, віднести їх до різних класів, дотримуючись достатнього ступеня деталізації, визначити інтерфейси класів, ієрархію їх наслідування та встановити ключові зв'язки між класами. Шаблон повинен, з одного боку, відповідати поставленим вимогам, а з другого – бути максимально загальним. Це найскладніше завдання в об'єктно-орієнтованому проєктуванні, адже при його вирішенні доводиться враховувати безліч факторів: інкапсуляцію, глибину деталізації, наявність залежностей, гнучкість, продуктивність, можливу еволюцію тощо.

Опрацювавши найпопулярніші патерни програмування, авторами було обрано архітектурний шаблон «MVC», але з деякими додатковими вимогами до функціональних елементів. Відтак система повинна мати єдиний контролер, завданням якого буде фільтрація вхідних даних та підключення базових інтерфейсів (класів).

Модель системи доцільно розділити на групи модулів, де кожен із елементів групи функціонуватиме за одним із принципів «UNIX» – один модуль (файл) виконує лише одну функцію. Для реалізації функціональності модулі можуть використовувати лише ті методи, що доступні в інтерфейсах (класах). Варто зауважити, що лише на рівні модулів можливе звернення до тих методів інтерфейсів, які передбачають отримання даних з бази даних та внесення змін до них.

Представлення формується системою шаблонізації на рівні модулів – кожний модуль формує певну частину представлення. Після завершення роботи логіки рівня модулів шаблонізатор формує цілісне представлення (вид) та повертає відповідь на запит (рисунк 4).

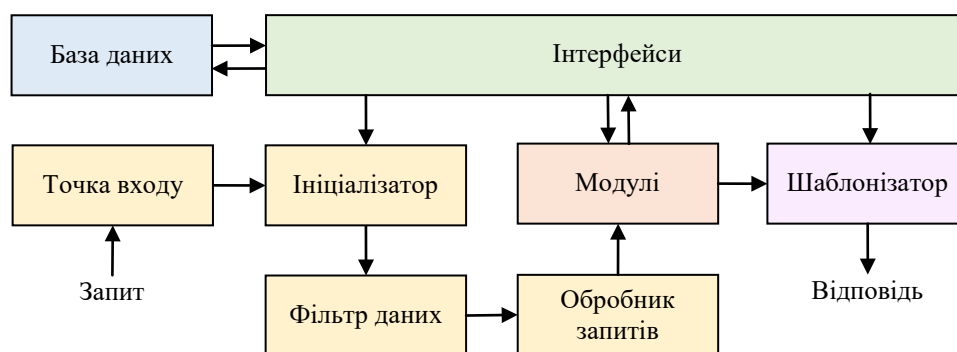


Рисунок 4 – Структура шаблону програмного засобу

Розробку додатка розпочнемо з огляду основних вимог і ключових принципів системи захисту, яка буде сформована на парадигмах моделі «Zero Trust» [10], що визначає повну відсутність довіри будь-кому та будь-яким даним.

Тобто згідно з однією з парадигм вищезазначеної моделі кожного разу, коли користувач звертається до системи, необхідно здійснювати повторну автентифікацію й авторизацію цього користувача (методи фактично різні – від перевірки логіну/пароля до динамічного токена). Кожна спроба доступу до системи розглядається як загроза доти, поки не доведено протилежне. Саме тут варто згадати раніше зазначену вимогу – вся система повинна мати єдину точку входу – це суттєво спрощує контроль походження даних та їх подальшу фільтрацію.

Як наслідок вищезазначеного – всі дані повинні піддаватися жорсткому фільтруванню та приведенню до необхідних типів з перевіркою їх поточного стану на адекватність та прийнятність. Ті дані, що фактично описують ідентифікатори записів з таблиць бази даних, повинні не тільки фільтруватися, а й перевірятися на валідність.

Враховуючи вищезазначені вимоги та правила, а також рекомендації стосовно написання «безпечного» коду [11], було прийнято рішення реалізувати фільтрацію вхідних даних та їх типізацію на рівні, який не буде залежати від функціонування окремих модулів системи (саме тому на рисунку 4 «фільтр даних» винесено в окремий блок). На етапі фільтрації та типізації перевіряється й цілісність даних, у разі виявлення певної нестандартної ситуації обробка поточного запису припиняється й система здійснює кваліфікацію рівня помилки – в разі підозри вторгнення система

блокує доступ, в іншому випадку користувачеві виводиться помилка.

Оскільки на етапі первинної фільтрації фактично неможливо надійно та достовірно перевірити адекватність і прийнятність даних, їх додаткова фільтрація та перевірка здійснюється вже безпосередньо на рівні модулів.

Підключення того чи іншого модуля здійснюється в обробнику запитів. У ході опрацювання вхідних даних приймається рішення стосовно формату відповіді («html» чи «JSON») та модуля, якому буде надано отримані дані та передано управління. Фактично «обробник запитів» (рисунку 4) є своєрідним «маршрутизатором», що здійснює перенаправлення потоку даних залежно від його властивостей.

Подальше опрацювання вхідних даних здійснюється вже тим чи іншим модулем. Як зазначалося раніше, тут же здійснюється й додаткова фільтрація даних, враховуючи їх специфіку. На рівні модулів здійснюється підключення до бази даних та задіюються необхідні методи щодо оперування даними в базі – від їх створення до видалення чи модифікації. Також одним із завдань, що покладається на рівень модулів, є організація керування рівнями доступу – безпосередньо в модулях перевіряються дозволи поточного користувача й визначається, які дії з інформацією є для нього можливими.

Функціонально в цьому програмному засобі модулі поділяються на два базові типи: модулі, що задіюються для керування відомостями про експертні провадження в системі, або так звані «редактори», і модулі отримання статистичних та аналітичних даних.

Завданням модулів керування даними є організація можливості створення та редагування як абсолютних, так і описових даних.

У ході реалізації цього типу модулів при оперуванні інформацією в СКБД варто застосовувати транзакції й, за можливості, всі алгоритми модифікації статистичних та допоміжних даних проводити лише на її рівні.

Тобто, наприклад, при оперуванні даними графа типу «Adjacency list» процедуру формування дерева батьківських записів (parents tree) та рівня вкладеності варто перенести на рівень СКБД – застосування тригерів «{BEFORE | AFTER} FOR EACH ROW» гарантує, що дані будуть адекватними та цілісними навіть при виникненні нестандартних ситуацій [14]. Окрім того, тригери мають суттєво більшу швидкість виконання порівняно з аналогічними алгоритмами опрацювання даних в додатку [19].

Другий тип модулів – статистичний – фактично цілком побудований на можливостях СКБД PostgreSQL формувати статистичні й аналітичні дані. Функціонально кожний такий модуль виконує певний запит до бази даних та повертає результат у необхідному форматі. Складність реалізації таких модулів полягає лише в складності формування запитів до бази даних й поєднання одного набору статистичних даних з іншими.

У ході реалізації функцій статистичних та аналітичних модулів було виявлено, що застосування конструкцій поєднання таблиць з метою виведення описових даних разом зі статистичними не є доцільним, оскільки значно ускладнює результуючий запит та сповільнює його виконання. Натомість ефективніше

вивантажити довідники в одномірні масиви типу «індекс – значення» й здійснювати формування результуючої відповіді вже після виконання запиту на отримання статистичних даних.

Також одним із завдань фактично будь-якого модуля є реалізація функцій фільтрування необхідних результуючих даних. Наприклад, модулі керування даними повинні містити функції пошуку записів за певну дату чи записів, що містять якісь певні дані. Статистичні модулі натомість мають забезпечувати можливість формування відомостей на певний момент у минулому чи за якимось окремим видом досліджень тощо.

Незалежно від виду модуля та його функцій, формування відповіді на запит здійснюється модулем за допомогою шаблонізатора – інтерфейсу, що дає змогу завантажувати файлові шаблони та реалізовує чіткий набір методів для заміни ключових міток (тегів) у цих шаблонах на необхідні дані. Шаблонами є файли з елементами html-коду, які фактично визначають візуальне подання тієї чи іншої інформації.

Такий підхід дає можливість уніфікувати зовнішній вигляд програмного засобу і визначити базові конструкції графічного інтерфейсу користувача (рисунок 5). Застосування однотипних елементів інтерфейсу дає змогу застосовувати для них загальні правила відображення, що значно покращує швидкодію додатка на стороні клієнта та візуальне сприйняття програмного засобу загалом [21].

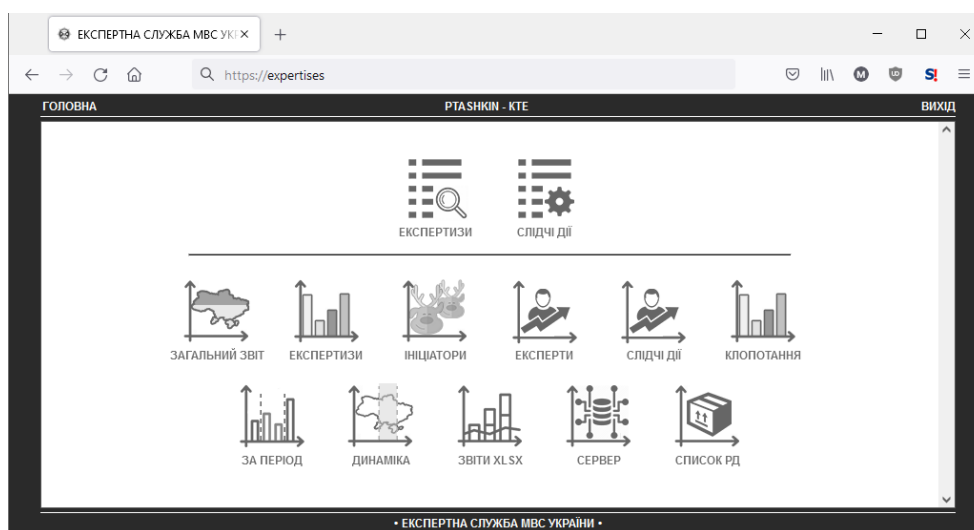


Рисунок 5 – Зовнішній вигляд програмного засобу у вікні браузера

Після закінчення етапу розробки було створено програмний засіб, який своїми функціональними можливостями дає змогу:

- здійснювати операції з відомостями про експертні провадження, а саме їх створення, модифікацію чи видалення з результируючим збереженням даних у СКБД PostgreSQL;

- здійснювати пошук та сортування відомостей про експертні провадження чи їх окремі елементи за будь-якою комбінацією пошукових параметрів;

- формувати статистичні й аналітичні дані (таблиці даних, графіки, діаграми) з відомостями про експертні провадження як в окремому підрозділі, так і в Експертній службі МВС України загалом за будь-якою комбінацією пошукових параметрів;

- здійснювати розділення доступу користувачів на рівні та визначати, який вид даних буде доступний певному користувачеві та які дії він зможе виконати з цими даними.

Результуюче програмне забезпечення побудоване за архітектурою «клієнт-сервер» з використанням лише безкоштовних технологій з відкритим вихідним кодом, а саме: Debian 10, Nginx 1.21, PHP 8.1, PostgreSQL 12.8. Окрім того, серверна частина додатка побудована з дотриманням парадигм моделі безпеки «Zero Trust» та здатна в автоматичному режимі реагувати на спроби несанкціонованого втручання в роботу. Серверна частина додатка реалізована за дещо модифікованою архітектурою «MVC» й не використовує жодної сторонньої бібліотеки чи фреймворку.

Проте найскладнішим етапом у створенні та забезпеченні функціонування програмного забезпечення є не стільки його створення, скільки його впровадження.

Процедуру *впровадження* інформаційно-аналітичної системи обліку експертних проваджень в Експертній службі МВС України запропоновано розділити в часі на три етапи:

- впровадження в одному підрозділі;
- впровадження в трьох підрозділах;
- впровадження в усіх підрозділах.

Кожен із вищезазначених етапів вимагає повернення до процесу розробки та адаптації функцій програмного засобу до особливостей функціонування того чи іншого підрозділу.

У ході впровадження інформаційно-аналітичної системи підтверджено правильність та ефективність використання web-технологій для розробки системи – програмний засіб однаково функціонує на різних операційних системах та вимагає від користувача лише браузер. Окрім того, значно спрощується процедура оновлення програмного забезпечення, адже, на відміну від desktop-ної архітектури, не потрібно здійснювати оновлення додатка в кожного клієнта, достатньо здійснити оновлення на сервері.

Наразі система успішно впроваджена в усіх підрозділах Експертної служби МВС України і використовується для ведення обліку експертних проваджень за такими напрямками судово-експертної діяльності, як: фототехнічні та портретні дослідження, дослідження відео- та звукозапису, телекомунікаційні дослідження, комп'ютерно-технічні дослідження, мистецтвознавчі дослідження, психологічні дослідження.

**Результати досліджень.** Аналізуючи проведені дослідження, можна стверджувати, що в цій роботі проведено аналіз проблемних питань, що виникають в одному з підрозділів Експертної служби МВС України, а саме – питання обліку експертних проваджень за напрямом комп'ютерно-технічних досліджень. У ході проведеного аналізу було сформовано проблематику, яка притаманна кожному підрозділу, та сформовано шлях вирішення цієї проблематики – створення власного програмного засобу.

В роботі в логічній послідовності висвітлено всі етапи проектування, розробки та впровадження інформаційно-аналітичної системи обліку експертних проваджень в Експертній службі МВС України.

Формуючи вимоги до майбутнього програмного забезпечення, було враховано не тільки специфіку напрямів дослідження чи окремі побажання до функціональних можливостей, а й сучасні норми та рекомендації щодо захисту інформації та високонавантаженого функціонування.

Враховуючи встановлені вимоги, було успішно розроблено програмне забезпечення, яке своїми функціональними можливостями

дає змогу необхідною мірою оперувати відомостями стосовно експертних проваджень і не тільки отримувати статистичні чи аналітичні дані, а й автоматизовано формувати звітність, що дає змогу суттєво спростити адміністративну роботу в кожному підрозділі Експертної служби МВС України.

Архітектура системи побудована за клієнт-серверним принципом з дотриманням парадигм моделі безпеки «Zero Trust». Розробка здійснена лише з використанням технологічних рішень з відкритим вихідним кодом і для своєї роботи зі сторони клієнта вимагає лише браузер.

З реалізованих статистичних та аналітичних можливостей варто виділити такі:

– будь-яку статистику та аналітику можливо формувати як на поточний час, так і на певний момент часу в минулому. Це дає можливість здійснювати порівняння даних з минулими періодами та аналізувати динаміку зміни показників діяльності;

– всі види статистики можливо фільтрувати за окремими видами досліджень чи їх групами. Така можливість дає змогу детальніше розуміти стан та проблемні питання окремого напрямку досліджень;

– можливість формування загальної статистики за всіма підрозділами Експертної служби МВС України – як за окремими напрямами досліджень, так і за їх групами чи взагалі за усіма напрямами. Ці функціональні можливості, насамперед, призначені для керівництва служби, оскільки дають змогу проаналізувати показники діяльності окремого підрозділу чи напряму на «загальному фоні»;

– в межах кожного підрозділу можливо формувати статистичні дані окремо по кожному виконавцю чи ініціатору судових експертиз та досліджень. Такі відомості дають можливість максимально точно розподіляти навантаження між виконавцями досліджень.

Функціональні можливості розробленого програмного засобу не є вичерпними та можуть бути доповнені необхідними функціями в майбутньому.

Розроблена система є першою інформаційно-аналітичною системою обліку експертних проваджень, що успішно впроваджена та використовується в усіх підрозділах Експертної служби МВС України.

**Обговорення результатів.** Фактично створена система є досить потужним знаряддям в руках управлінця, який може здійснити аналіз отриманих відомостей. Адже саме наявність доступу до статистичних та аналітичних даних дає змогу вчасно та достовірно здійснювати необхідні коригувальні дії для забезпечення успішного функціонування підрозділу в цілому.

Як вже зазначалося, цей програмний засіб дає можливість формувати статистичні та аналітичні дані за будь-якими параметрами як у реальному часі, так і на момент часу в минулому. Однак, окрім цього, система здатна здійснювати прогнозування показників на момент часу в майбутньому.

Реалізовані алгоритми прогнозування дають змогу аналізувати необхідні статистичні дані за періоди часу минулих років і здійснювати їх порівняння з аналогічними періодами часу поточного року. В ході опрацювання вираховується тенденція зміни даних та її накладення на аналогічні періоди в майбутньому, але з урахуванням базової річної зміни.

Наприклад, аналізуючи статистичні дані, було встановлено, що різке збільшення надходжень експертних проваджень до підрозділів Експертної служби МВС України припадає лише декільком місяцям року. А враховуючи щорічну тенденцію зміни кількості експертних проваджень, можна здійснити прогнозування кількісних показників як на кінець поточного року, так і на певний момент часу в майбутньому.

Звичайно, вести мову про поняття точності таких прогнозів наразі зарано, оскільки система має досить малий набір даних. Проте подальше використання системи зумовлюватиме збільшення набору даних, що сприятиме зростанню точності таких прогнозів.

**Висновки.** Перенесення інформації про діяльність підприємств чи організацій у цифровий простір є процесом неминучим. В цифровому вигляді інформацію значно простіше зберігати та опрацьовувати. Маючи набори даних у цифровому вигляді, їх можливо багаторазово опрацьовувати та аналізувати, формуючи статистичні й аналітичні відомості залежно від завдань сьогодення. Наразі досить складно вчасно та правильно здійснювати будь-яку управлінську діяльність, не маючи

доступу до різнопланової, актуальної та достовірної інформації, збір і надання якої ґрунтуються на можливостях сучасних спеціалізованих інформаційно-аналітичних систем.

У роботі здійснено аналіз вимог до програмного забезпечення, що може бути використане в підрозділах Експертної служби МВС України. Фактично до програмних засобів, які будуть використовуватися для обробки інформації, витік якої може справити негативний вплив на життєво важливі інтереси людини і громадянина, суспільства та держави в цілому, висувається ряд вимог стосовно забезпечення належного рівня захисту.

Головним завданням роботи було послідовне та логічне висвітлення всіх етапів проектування, реальної розробки та впровадження власної інформаційно-аналітичної системи обліку експертних проваджень в Експертній службі МВС України.

*Наукова новизна* дослідження полягає у побудові моделей структури бази даних та інформаційно-аналітичної системи обліку експертних проваджень в цілому, організації даних і шаблону програмного засобу, що дають змогу проектувати та створювати сучасні програмні засоби для збереження відомостей про експертні провадження, їхньої автоматичної обробки, формування звітності, статистичних і аналітичних даних.

*Практична значущість* отриманих результатів ґрунтується на створенні функціонально закінченої інформаційно-аналітичної системи та її успішному впровадженні в діяльність кожного підрозділу Експертної служби МВС України. Наразі програмний засіб використовується в повсякденній роботі працівниками вищезазначеної служби за напрямками фототехнічних, портретних, комп'ютерно-технічних, мистецтвознавчих, психологічних та телекомунікаційних досліджень, а також досліджень відео- та звукозапису.

На цьому етапі функціонування інформаційно-аналітична система не тільки дає можливість зберігати дані, керувати ними чи здійснювати пошук серед них, а й має потужні алгоритми формування звітності, статистичної та аналітичної інформації. Така функціональність дає змогу працівникам окремих напрямів Експертної служби МВС України здійснювати

значно ефективнішу управлінську діяльність та своєчасно корегувати діяльність в окремих напрямках чи підрозділах.

Окрім того, в тестовому режимі функціонує система прогнозування показників діяльності, алгоритми якої базуються на детальному аналізі наявних даних, їх сегментації в часі та визначення параметрів, закономірностей та динаміки їх зміни. Такі функції покликані спростити адміністративну діяльність, що пов'язана з плануванням діяльності в майбутньому.

Створена інформаційно-аналітична система придатна для подальшої її модифікації та має перспективу для проведення подальших досліджень, а саме за напрямом розроблення та реалізації нових алгоритмів аналізу даних й формування більш точної аналітичної інформації.

Розроблену систему можна віднести до інформаційних систем керування підприємством чи установою загалом, але в неї закладені лише функції обліку експертних проваджень за певними напрямками та аналізу кількісних показників діяльності підрозділів Експертної служби МВС України, без втручання в інші аспекти їх функціонування.

### Список використаних джерел

- [1] МВС України. (2009, жовт. 12). *Наказ № 436, Про затвердження Положення про Інтегровану інформаційно-пошукову систему органів внутрішніх справ України*. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/z1256-09/conv#Text>
- [2] Кабінет Міністрів України. (2018, листоп. 14). *Постанова № 1024, Про затвердження Положення про єдину інформаційну систему Міністерства внутрішніх справ та переліку її пріоритетних інформаційних ресурсів*. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/1024-2018-%D0%BF/>
- [3] Проект Стратегії кібербезпеки України на 2021–2025 роки. [Електронний ресурс]. Режим доступу: [https://www.rnbo.gov.ua/files/2021/STRATEGIYA%20KYBERBEZPEKI/proekt%20strategii\\_kyberbezpeki\\_Ukr.pdf](https://www.rnbo.gov.ua/files/2021/STRATEGIYA%20KYBERBEZPEKI/proekt%20strategii_kyberbezpeki_Ukr.pdf)

- [4] Ю. В. Триус, Г. О. Заспа, О. С. Коже-м'якін, та А. В. Аширова, "Інформаційно-аналітична система підтримки освітньої діяльності структурних підрозділів закладів вищої освіти", *Вісник Черкаського державного технологічного університету*, № 4, с. 27-38, 2020.
- [5] Н. В. Бугас, та О. О. Коваленко, "Інформаційна система як умова ефективних управлінських рішень", *Ефективна економіка*, № 12, 2016. [Електронний ресурс]. Режим доступу: <http://www.economy.nauka.com.ua/?op=1&z=5313>
- [6] М. Клеппман, *Высоконагруженные приложения. Программирование, масштабирование, поддержка*. Санкт-Петербург, Россия: Питер, 2018. ISBN: 978-5-4461-0512-0.
- [7] D. DeJonghe, *NGINX Cookbook: Advanced Recipes for High Performance Load Balancing*. CA, USA: O'Reilly Media, 2019. ISBN: 978-1-491-96893-2.
- [8] Л. Кэмпбелл, и Ч. Мейджорс, *Базы данных. Инжиниринг надежности*. Санкт-Петербург, Россия: Питер, 2020. ISBN 978-5-4461-1310-1.
- [9] Д. Грас, *Data Science. Наука о данных с нуля*: 2-е изд. Санкт-Петербург, Россия: БХВ-Петербург, 2021. ISBN: 978-5-9775-6731-2.
- [10] J. Garbis, and J. Chapman, *Zero Trust Security: An Enterprise Guide*. Boston, MA, USA: Apress. ISBN-13: 978-1-4842-6701-1.
- [11] Д. Джонсон, Д. Деоган, и Д. Савано, *Безопасно by design*. Санкт-Петербург, Россия: Питер, 2021. ISBN 978-5-4461-1507-5.
- [12] Б. Новиков, Е. Горшкова, и Н. Графеева, *Основы технологий баз данных: учеб. пособ.* Москва, Россия: ДМК Пресс, 2020. ISBN 978-5-97060-841-8.
- [13] Д. Босуелл, и Т. Фаучер. *Читаемый код или программирование как искусство*. Санкт-Петербург, Россия: Питер, 2012. ISBN: 978-5-459-01188-3.
- [14] Г. Шениг, *PostgreSQL 11. Мастерство разработки*. Пер. с англ. А. А. Слинкина. Москва, Россия: ДМК Пресс, 2019. ISBN: 978-5-97060-671-1.
- [15] Н. Dombrovskaya, В. Novikov, and А. Baillieкова, *PostgreSQL Query Optimization: The Ultimate Guide to Building Efficient Queries*. Chicago, IL, USA: Apress, 2021. ISBN: 978-1-4842-6884-1.
- [16] PostgreSQL 11.12 Documentation. [Online]. Available: <https://www.postgresql.org/docs/11/index.html>
- [17] Э. Гамма, Р. Хелм, Р. Джонсон, и Дж. Влиссидес, *Паттерны объектно-ориентированного проектирования*. Санкт-Петербург, Россия: Питер, 2020. ISBN 978-5-4461-1595-2.
- [18] С. Pitt, *Pro PHP 8 MVC: Model View Controller Architecture-Driven Application Development*. Cape Town, South Africa: Apress, 2021. ISBN: 978-1-4842-6956-5.
- [19] М. Zandstra, *PHP 8 Objects, Patterns, and Practice: Mastering OO Enhancements, Design Patterns, and Essential Development Tools*. Brighton, UK: Apress, 2021. ISBN: 978-1-4842-6790-5.
- [20] Руководство по PHP. [Электронный ресурс]. Режим доступа: <https://www.php.net/manual/ru/>
- [21] К. Татро, и П. Макинтайр, *Создаем динамические веб-сайты на PHP*. 4-е междунар. изд. Санкт-Петербург, Россия: Питер, 2021. ISBN: 978-5-4461-1488-7.

## References

- [1] The Ministry of Internal Affairs of Ukraine. (2009, Oct. 12). *Order No. 436, On approval of the Regulations on the Integrated Information Search System of the Internal Affairs Bodies of Ukraine*. [Online]. Available: <https://zakon.rada.gov.ua/laws/show/z1256-09/conv#Text> [in Ukrainian].
- [2] The Cabinet of Ministers of Ukraine. (2018, Nov. 14). *Resolution No. 1024, On approval of the Regulation on the unified information system of the Ministry of Internal Affairs and the list of its priority information resources*. [Online]. Available: <https://zakon.rada.gov.ua/laws/show/1024-2018-%D0%BF/conv#Text> [in Ukrainian].

- [3] Draft of the Cybersecurity Strategy of Ukraine (2021–2025). [Online]. Available: [https://www.rnbo.gov.ua/files/2021/STRATEGIYA%20KYBERBEZPEKI/proekt%20strategii\\_kyberbezpeki\\_Ukr.pdf](https://www.rnbo.gov.ua/files/2021/STRATEGIYA%20KYBERBEZPEKI/proekt%20strategii_kyberbezpeki_Ukr.pdf) [in Ukrainian].
- [4] Yu. V. Tryus, H. O. Zaspа, O. S. Kozhemiakin, and A. V. Ashyrova, "Information and analytical system for educational activities support of structural divisions of higher education institutions", *Visnyk Cherkaskogo derzhavnogo tehnologichnogo universitetu*, no. 4, pp. 27–38, 2020 [in Ukrainian].
- [5] N. V. Bugas, and O. O. Kovalenko, "Information system as a condition for effective management decisions", *Efektivna ekonomika*, no. 12, 2016. [Online]. Available: <http://www.economy.nayka.com.ua/?op=1&z=5313> [in Ukrainian].
- [6] M. Kleppmann, *Highly Loaded Applications. Programming, Scaling, Support*. St. Petersburg, Russia: Piter, 2018. ISBN: 978-5-4461-0512-0 [in Russian].
- [7] D. DeJonghe, *NGINX Cookbook: Advanced Recipes for High Performance Load Balancing*. CA, USA: O'Reilly Media, 2019. ISBN: 978-1-491-96893-2.
- [8] L. Campbell, and Ch. Majors, *Databases. Reliability Engineering*. St. Petersburg, Russia: Piter, 2020 [in Russian]. ISBN 978-5-4461-1310-1.
- [9] D. Grus, *Data Science. Data Science from Scratch: 2<sup>nd</sup> Ed.* St. Petersburg, Russia: BHV-Peteburg, 2021 [in Russian]. ISBN: 978-5-9775-6731-2.
- [10] J. Garbis, and J. Chapman, *Zero Trust Security: An Enterprise Guide*. Boston, MA, USA: Apress. ISBN-13: 978-1-4842-6701-1.
- [11] D. Johnson, D. Deogun, and D. Sawano, *Secure by Design*. St. Petersburg, Russia: Piter, 2021 [in Russian]. ISBN 978-5-4461-1507-5.
- [12] B. Novikov, E. Gorshkova, and N. Grafeeva, *Database Technology Fundamentals: A tutorial*. Moscow, Russia: DMK Press, 2020. [in Russian]. ISBN 978-5-97060-841-8.
- [13] D. Boswell, and T. Foucher, *Readable code or programming as art*. St. Petersburg, Russia: Piter, 2012 [in Russian]. ISBN: 978-5-459-01188-3.
- [14] G. Schönig, PostgreSQL 11. *Mastering of Development*. Transl. from Eng. by A. A. Slinkin. Moscow, Russia: DMK Press, 2019 [in Russian]. ISBN: 978-5-97060-671-1.
- [15] H. Dombrovskaya, B. Novikov, and A. Bailliekova, *PostgreSQL Query Optimization: The Ultimate Guide to Building Efficient Queries*. Chicago, IL, USA: Apress, 2021. ISBN: 978-1-4842-6884-1.
- [16] PostgreSQL 11.12 Documentation. [Online]. Available: <https://www.postgresql.org/docs/11/index.html>
- [17] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Object-Oriented Design Patterns*. St. Petersburg, Russia: Piter, 2020 [in Russian]. ISBN 978-5-4461-1595-2.
- [18] C. Pitt, *Pro PHP 8 MVC: Model View Controller Architecture-Driven Application Development*. Cape Town, South Africa: Apress, 2021. ISBN: 978-1-4842-6956-5.
- [19] M. Zandstra, *PHP 8 Objects, Patterns, and Practice: Mastering OO Enhancements, Design Patterns, and Essential Development Tools*. Brighton, UK: Apress, 2021. ISBN: 978-1-4842-6790-5.
- [20] PHP documentation. [Online]. Available: <https://www.php.net/manual/ru/>
- [21] K. Tatroe, and P. Macintyre, *Creating dynamic websites in PHP. 4<sup>th</sup> Int. Ed.* St. Petersburg, Russia: Piter, 2021 [in Russian]. ISBN: 978-5-4461-1488-7.

**R. L. Ptashkin<sup>1</sup>,**

e-mail: ndekc.ck@gmail.com

**A. V. Honcharov<sup>2</sup>,** *Ph. D., Associate Professor,*

e-mail: a.honcharov@chdtu.edu.ua

**O. S. Havrysh<sup>2</sup>,** *Ph. D., Associate Professor*

e-mail: o.havrysh@chdtu.edu.ua

<sup>1</sup>Cherkasy Scientific Research Forensic Centre of the Ministry of Internal Affairs of Ukraine  
Pasterivska st., 104, Cherkasy, 18000, Ukraine

<sup>2</sup>Cherkasy State Technological University  
Shevchenko Blvd., 460, Cherkasy, 18006, Ukraine

### **INFORMATION AND ANALYTICAL SYSTEM FOR ACCOUNTING OF EXPERT PROCEEDINGS IN DEPARTMENTS OF THE EXPERT SERVICE OF THE MINISTRY OF INTERNAL AFFAIRS OF UKRAINE**

*This paper considers the experience of modelling, development and implementation of new software, such as information and analytical system, for examination in government forensics centres. The authors have made analysis of current requirements to software which can be used in government structures. The list of basic requirements and most important principles of software development has been made. After analysing business processes in forensics centres the authors have made the list of needed functions and facilities for future software.*

*One of problems consists in the choice of basic technology for software development. The authors have analysed the most popular technologies and chosen web-oriented architecture – clients will be connecting to software using browser for see graphical user interface, same as to regular website. All business logic of software will be provided at server side.*

*During the development, the authors have paid special attention to the issue of information protection. Cybersecurity is one of the main requirements for software, which can be used in divisions of the Ministry of Internal Affairs of Ukraine. "Zero Trust" model has been chosen as the main model security for future software. This means that internal algorithms must scan and filter any incoming data in few levels. The system must have autonomic system of reacting to attacks.*

*As key technologies the PostgreSQL as database management system, NGINX as web-server software, PHP programming language for creating server-side software and HTML/CSS/Java Script for making graphical user interface have been chosen. Based on security requirements, no third-party libraries or frameworks were used in the development.*

*Directly for the implementation of the server part of the system, the authors have chosen a slightly modified MVC model and implemented the principle of absolute modularity of the system.*

*After the development, the information and analytical system has been successfully implemented in the activities of all divisions of the Expert Service of the Ministry of Internal Affairs of Ukraine. Currently, the system is actively used for automated reporting and generation of statistical and analytical data on certain activities of the service.*

**Keywords:** *information and analytical system, data analysis, statistics, analytics, software development, databases, web, PHP, PostgreSQL.*