



О. О. Корнак, Ю. І. Грицюк

Національний університет "Львівська політехніка", м. Львів, Україна

ПРОЕКТУВАННЯ СИСТЕМИ УПРАВЛІННЯ КОШТАМИ БЛАГОДІЙНОГО ФОНДУ

Спроектовано систему управління коштами благодійного фонду, яка забезпечить взаємодію учасників процесу в окремій інформаційній системі задля надання їм актуальної, достовірної та цілісної інформації на етапах отримання коштів, їх розподілу та аналізу використання. З'ясовано, що відповідна інформаційна система для організації благодійного процесу має містити методи і способи збору, нагромадження, реєстрації, передавання, відображення, зберігання, пошуку, модифікації, аналізу, захисту та видачі необхідної інформації всім зацікавленим сторонам. Проаналізовано найпопулярніші сайти та платформи України для роботи благодійних фондів, що допомогло краще зрозуміти функціональність системи управління коштами організації та їх основного призначення. Визначено основне архітектурне рішення, яке використано для побудови веб-додатку, а саме – шаблон MVC, який застосовують як для проектування архітектури програмного засобу, так і для його розроблення.

Побудовано діаграму варіантів використання системи управління коштами, що показує усіх користувачів системи, а також ту її функціональність, доступ до якої має відповідний клас користувачів. Для відображення динамічної роботи веб-додатку було побудовано діаграму послідовності, яка відтворює процес виділення коштів з бюджету фонду для лікування нужденних осіб. Спроектовано базу даних веб-додатку, яка передбачає виконання таких етапів: проектування на концептуальному рівні з використання ER-моделі; проектування на логічному рівні, внаслідок якого було побудовано реляційну базу даних; проектування на фізичному рівні в межах відповідної СУБД, для якої було розроблено відповідну базу даних. Розроблено графічний інтерфейс користувача: головні його елементи згідно з відомими принципами їх побудови; визначено загальну структуру та основне сприйняття його вигляду. На підставі запропонованих рішень розроблено усі інші компоненти, необхідні для роботи веб-додатку, графічні елементи та сторінки.

Ключові слова: інформаційна система; веб-додаток; архітектурне рішення; діаграма варіантів використання; діаграма послідовності; реляційна база даних.

Вступ

У людській спільноті часто виникають різноманітні труднощі – будь-то соціальні, екологічні чи навіть економічні. Але світ не без добрих людей, які готові в той чи інший спосіб надати допомогу, а інколи й навіть запобігти виникненню скрутних ситуацій [30]. Найшвидший та, мабуть, найпростіший спосіб зробити свій внесок у вирішення певних проблем – допомогти нужденним фінансово.

Існує чимало соціальних платформ і різноманітних фондів, робота яких спрямована на благодійну діяльність [28]. Благодійні фонди, наприклад, які допомагають хворим дітям, надзвичайно важливі насамперед в нашій державі, де місячний дохід громадян та одноразові ціни на ліки мало чим різняться між собою. Тому для забезпечення хворих необхідними медикаментами їхні опікуни часто звертаються за допомогою до подібних фондів. Головним завданням діяльності благодійних фондів є збір коштів, їх нагромадження та розподіл між нужденними [34, 44]. Тут вкрай важливо, щоб інформація про наявність таких коштів завжди була актуальною та доступною опікуну в будь-який момент часу.

На сьогодні жодна галузь людської діяльності не обходиться без інформаційних технологій [16, 22, 35]. Однак, тільки деякі благодійні фонди, де значна частина працівників/волонтерів зосереджені на обліку надходжень, розподілі та звітності про витрати, всі ці та багато інших операцій здійснюють за допомогою відповідних інформаційних систем [33, 36, 43]. Тому для ефективної діяльності таких фондів важливо, щоб у них була можливість використовувати відповідні програмні засоби, які б давали їм змогу надавати інформацію про їхній, наприклад, бюджет чи витрати як його керівництву (наглядовій раді), так і волонтерам. Також така інформаційна система має формувати джерела надходження коштів, реєструвати нові звернення нужденних і розподіляти зібрані кошти між ними в міру їх наявності. Для якісного вирішення зазначених дій необхідно, щоб цей програмний засіб забезпечував ефективне управління коштами благодійного фонду, а зосереджена в ньому інформація завжди була актуальною та постійно обновлюваною, що дасть змогу нужденним у скрутний момент скористатися посиленою допомогою фонду [30].

Водночас, оскільки кошти до фонду надходять з різ-

Інформація про авторів:

Корнак Олена Олексіївна, студент, кафедра програмного забезпечення. Email: olenkakornak@gmail.com

Грицюк Юрій Іванович, д-р техн. наук, професор, кафедра програмного забезпечення.

Email: yurii.i.hrytsiuk@lpnu.ua; <https://orcid.org/0000-0001-8183-3466>

Цитування за ДСТУ: Корнак О. І., Грицюк Ю. І. Проектування системи управління коштами благодійного фонду. Науковий вісник НЛТУ України. 2021, т. 31, № 2. С. 121–132.

Citation APA: Kornak, O. I., & Hrytsiuk, Yu. I. (2021). Designing a fund management system for a charity fund. *Scientific Bulletin of UNFU*, 31(2), 121–132. <https://doi.org/10.36930/40310220>

них джерел, назви яких і обсяги пожертв заносять в облікову базу даних системи, то вона має бути доступною широкій громадськості і, що найважливіше, цілісною, щоб зловмисники не змогли її спотворити, а також спостережною, де фіксуватимуться всі дії її користувачів [28]. Адже не секрет, що можуть виявитися працівники чи волонтери фонду, які захочуть скористатися його коштами без відома його керівництва, тобто поза межами благодійного процесу.

Головними учасниками благодійного процесу мають бути: благодійники, фінансисти чи пересічні громадяни; управителі благодійних коштів; нужденні благодійних коштів [30]. Кожний учасник цього процесу має мати доступ до бази даних системи для роботи з нею шляхом розмежування доступу (англ. *Access Mediation*) – сукупність процедур, що реалізують перевірку запитів на доступ і оцінку відповідної операції на можливість надання доступу [3, 4]. При використанні розмежування доступу кожному учаснику (ролі) вказують набір повноважень, що представляють йому(їй) набір прав доступу до відповідних об'єктів інформаційної системи. Повноваження призначають ролям відповідно до принципу найменших привілеїв, з якого випливає, що кожний користувач повинен мати тільки мінімально необхідні для виконання своєї роботи повноваження [25].

З огляду на зазначене вище, спроектована система управління коштами благодійного фонду має передбачати не тільки вільний доступ до її відкритої інформації, але й враховувати політику інформаційної безпеки, що є актуальним і вкрай важливим завданням кожного програмного продукту, розміщеного в мережі Інтернет у вигляді будь-якого сайту чи сервісу.

Об'єкт дослідження – система управління коштами благодійного фонду.

Предмет дослідження – методи і засоби проектування системи управління коштами благодійного фонду, що дасть змогу сформувати взаємодію учасників благодійного процесу залежно від наданих їм прав доступу до системи.

Мета роботи – розробити систему управління коштами благодійного фонду, яка б давала змогу сформувати взаємодію учасників процесу в окремій інформаційній системі задля забезпечення їх актуальною, достовірною та цілісною інформацією на етапах отримання коштів, розподілу та аналізу використання.

Для реалізації зазначеної мети потрібно вирішити такі *основні завдання дослідження*:

- 1) проаналізувати роботу благодійних фондів, з'ясувати особливості застосування системи управління їх коштами;
- 2) визначити основні завдання системи управління коштами благодійного фонду, встановити її структуру компонент і функціональні можливості;
- 3) розробити специфікацію вимог до програмного додатку, побудувати його архітектуру та навести перелік інтерфейсів користувача;

Наукова новизна отриманих результатів дослідження – розроблено систему управління коштами благодійного фонду, яка забезпечує взаємодію учасників процесу в інформаційній системі, а також надає їм актуальну, достовірну та цілісну інформацією на всіх етапах благодійного процесу.

Практична значущість результатів дослідження – спроектовану систему управління коштами благодійно-

го фонду після її програмної реалізації можна використовувати у відповідних установах для підтримки взаємодії учасників благодійного процесу залежно від наданих їм прав доступу до системи.

Аналіз сучасних програмних засобів – аналогів проектованої системи. З кожним днем благодійність чи допомога у вирішенні навіть буденних проблем стає все більш поширеною та популярною [14, 30, 34]. Із зростанням попиту на благодійність ІТ-фахівці створюють в мережі Інтернет все нові і нові сторінки чи сайту для організації благодійного фонду. Після ознайомлення з їхньою інформацією і будучи небайдужою до неї, кожна людина може зробити свій фінансовий вклад для вирішення певних проблем – бездомних тварин, людей – потерпілих чи нужденних, ініціатив громадськості певної місцевості тощо. Адже серед буденної метушні, де кожен зайнятий своєю справою, це надзвичайно зручний та швидкий спосіб надати посильну допомогу іншим, не встаючи зі свого робочого місця [44].

В Україні серед найпопулярніших сайтів та платформ, які зайняті благодійністю, можна виділити такі: платформа Про dobro.ua, сайт Let's help!, сайт благодійного фонду "Кожен може". Проаналізуємо їхні можливості дещо детальніше.

Платформа Про dobro.ua – це українська біржа благодійності (рис. 1,а), яка підтримує роботу найбільшої в Україні фандрейзингової платформи, що залучає кошти на різноманітні благодійні та соціальні проекти [43]. Нагадаємо, що фандрейзинг (англ. *Fundraising* – збір коштів) – процес залучення коштів та інших ресурсів (людських, матеріальних, інформаційних тощо), які організація не може забезпечити самостійно, однак вони є необхідними для реалізації певного проекту або своєї діяльності загалом.

Використовуючи цей програмний продукт, кожен користувач може розпочати збір коштів на будь-який зі своїх проектів. Також на сайті можна відстежувати хід збору коштів на наявні проекти з різних сфер життя. Серед основних функцій платформи можна виділити такі [43]:

- доступ користувачів до інформації про зареєстровані проблеми та надання доступу тим з них, що бажають допомогти фінансово через засіб для оформлення платежу;
- перегляд користувачами відповідного сайту донорів проектів;
- оформлення SMS-благодійності;
- створення кампаній для збору коштів для проектів;
- поповнення скарбнички платформи для допомоги у критичний момент;
- можливість реєстрації особистого кабінету користувача, де буде розміщена фінансова інформація та деталі підтримки проектів з ходом їхньої реалізації;
- створення заходів (концертів чи виставок), де користувачі можуть долучитися до їхнього проведення. У такий спосіб усі зібрані кошти на даних заходах будуть спрямовані на благодійність.

Сайт Let's help – це міжнародний благодійний фонд (рис. 1,б), основними напрямками діяльності якого є: допомога самотнім літнім і хворим особам, дітям-сиротам [36]. За інформацією, розміщеною на сайті фонду, серед основних його місій можна виділити такі:

- розвиток традицій благодійності в Україні;
 - об'єднання людей у спільноту;
 - надихання на добрі справи.
- До основних функцій даного сайту належать:
- отримання інформації про проекти фонду та можливості оформлення грошового переказу для їх підтримки;

- перегляд донорів кожного окремого проекту;
- створення нових заходів для збору коштів;

- оформлення кабінету волонтера на сайті;
- отримання доступу до звітів роботи фонду.



Рис. 1. Інтерфейси сайтів: а) Про dobro.ua; б) Let's help!; в) Кожен може

Сайт "Кожен може" – це благодійний фонд, основною місією якого є освіта, розвиток культури благодійності та волонтерства, становлення відповідального та толерантного суспільства в Україні, а також допомога нужденним [33]. Основними напрямками роботи фонду є допомога:

- дітям з інвалідністю, з родин, що постраждали від війни на Сході України;
- важкохворим самотнім пенсіонерам, що постраждали від воєнних дій;
- медичним обладнанням підопічних дитячих лікарень в Україні.

До основних функцій сайту можна віднести:

- перегляд інформації про різні проекти та оформлення фінансової допомоги;
- отримання звітів щодо діяльності благодійного фонду;
- отримання інформації щодо поточної роботи благодійного фонду.

Проте, в Україні існують також і благодійні фонди, які допомагають багатьом нужденним, не маючи для цього ні сторінок, ні сайтів у мережі Інтернет. Тому необхідно, щоб кожен благодійний фонд мав у своєму розпорядженні відповідне ПЗ, використання якого давала б змогу створювати нові проекти для надання допомоги, формувати бюджет фонду через різні джерела фінансування та розподіляти кошти між нужденними [14]. Така робота фонду стане більш ефективною та

наглядною, адже уся актуальна інформація завжди буде доступною, а усі платіжні операції можна буде проводити значно швидше у будь-який час. Окрім цього, звіти про діяльність благодійного фонду й статистику, на підставі якої керівництво фонду буде робити відповідні висновки, можна формувати автоматично та оновлювати на підставі інформації, збереженої в базі даних системи [25].

Результати дослідження та їх обговорення

Опис предметної області знань. Благодійність – це добровільна особиста або майнова допомога для досягнення встановлених цілей [14, 44]. Як один з проявів гуманності людини, благодійність набуває все більшої популярності в усьому світі [30]. Люди переймаються екологічними чи економічними проблемами певної місцевості, регіону чи навіть країни, але найбільше співчужають негараздам чи скрутним становищам інших людей. Тому вони часто готові надати власні кошти для підтримки того, хто потребує нагальної допомоги [34].

Існує багато платформ та сайтів, які організують благодійний процес [33, 36, 43]. Вирішуючи ту чи іншу проблему, адміністратори таких програмних додатків розміщують на своїх сторінках усю необхідну інформацію. Кожна юридична чи фізична особа може ознайомитися із різними благодійними проектами та, за бажанням, допомогти фінансово, не виходячи з дому [4].

Загалом, зовнішня діяльність благодійного фонду – це щоденна професійна робота волонтерів щодо забезпечення потреб отримувачів допомоги, пошуку донорів для її надання. Це робота з залучення юридичних і фізичних осіб, що можуть пожертвувати свої кошти, з забезпечення документообігу, а також подачі звітності в органи фіскальної служби та на вимогу донорам фонду [28]. Фонд – це певний посередник між нужденними та закладами, що надають їм допомогу [34]. Маючи власний бюджет, фонд наповнює його з різних джерел і на замовлення нужденного, зареєстрованого у системі, виділяє йому кошти. Існують фонди, що 20 % благодійних внесків спрямовують на власні адміністративні витрати, але більшість фондів усі виручені чи отримані кошти спрямовують виключно на благодійність, тобто, на допомогу нужденним [14].

Є чимало способів для збору коштів на благодійність, де небайдужі особи можуть здійснити пожертви, найпоширенішими серед яких є такі [32]:

- *благодійний вечір* – участь в заході можуть брати усі зацікавлені особи; кошти отримують за рахунок продажу квитків та особистих пожертв гостей;
- *святкування різних подій* – визначають притаманну певній місцевості подію, наприклад, тематичний ярмарок ручних виробів, зустрічі з письменниками чи меценатами тощо, де можна назбирати необхідну суму для певної мети;
- *розпродаж речей* – аукціон особистих речей відомих людей – спортсменів, політиків чи колекціонерів, що вони їх пожертвували на благодійність.

Використовуючи будь-які зі способів отримання коштів, керівництво фонду може залучати нових донорів, що зацікавлені у вирішенні певних проблем. Бюджет фонду також можна збільшувати за рахунок пожертв від юридичних осіб чи отриманні гранту від гуманітарних організацій [34].

Сформувавши бюджет фонду, його керівництво ділить певну частку коштів між нужденними, що звернулися за допомогою [28]. Передусім, з'ясовують суму зібраних коштів, необхідних для реалізації того чи іншого проекту. Якщо після завершення терміну збору коштів буде виявлено, що необхідну суму не отримано, то благодійний фонд жертвує кошти із власного бюджету. Однак, ці кошти виділяють нужденному не на його рахунок, а на рахунок організації, будь-то лікарня чи інша установа, яка буде надавати йому безпосередню допомогу.

Водночас, внутрішня діяльність будь-якого фонду – це щоденна робота, зазвичай, бухгалтерів з фінансовими потоками даних – трансакцій, рахунків і звітності. Вона є важливою складовою для управління благодійним фондом, адже містить інформацію про стан його грошових потоків – приходу коштів і їх витрат [25]. Фінансову інформацію можна використовувати для обліку, контролю та аналізу роботи фонду, тому вона завжди має відповідати таким властивостям [22]: достовірність та повнота; цінність та актуальність; ясність та зрозумілість.

Автоматизована інформаційна система – це один із способів забезпечення фінансової інформації заданими властивостям, позаяк організовує процес збереження та оброблення даних задля задоволення інформаційних потреб різних користувачів системи [10, 35]. Однак, для ефективного використання фінансової інформації необхідно виконати деякі етапи: організація збору даних і їх початкове оброблення, здійснення передачі даних, їх

зберігання та нагромадження, забезпечення перетворення та видачу даних, а також уможливлення вільного доступу до них різних користувачів [17, 22].

Документування фінансових даних з використанням інформаційних систем забезпечує значні переваги для роботи благодійного фонду, а саме [3, 16]:

- отримання повних і достовірних даних;
- підвищення якості облікових робіт;
- збільшення швидкодії виконання операцій збирання, оброблення та введення даних;
- вчасне надання керівництву фонду необхідної інформації для прийняття рішень.

Для благодійного фонду фінансова інформація висвітлює надійний облік коштів і прозорість проведення фінансових операцій [16, 26]. Поточний стан фінансів формують у вигляді звітів, що подають у органи фіскальної служби, а також у вигляді відповідних статистичних даних для керівництва фонду.

Отже, благодійний фонд – організація, яка має власний бюджет, наповнює його з різних джерел доходу, розподіляє кошти між нужденними, документує весь обіг коштів для створення звітів [14]. Тому кожна небайдужа особа, що пожертвувала власні кошти фонду, чи внесла на особистий рахунок нужденного може бути впевнена, що вони пішли на благодійність [34]. Для держави – це доказ того, що фонд працює в рамках чинного законодавства, позаяк всі отримані від донорів кошти дійсно спрямовані на добрі справи.

Очікуваний результат реалізації проекту. Дослідження предметної області – діяльності благодійних фондів, аналіз наявних способів наповнення їхнього бюджету, можливі варіанти розподілу коштів між нужденними – все це і багато іншого дало змогу визначити основні особливості роботи відповідної інформаційної системи [23]. Також було встановлено головні інформаційні процеси та операції, що потребують автоматизації виконання за допомогою цієї систем [16]. Отже, як наслідок проектування системи управління коштами благодійного фонду, передбачаємо отримання таких результатів [23]:

- розмежування доступу користувачів до системи;
- формування бюджету фонду коштами з визначених джерел доходу;
- наповнення системи особами, що звернулися за допомогою;
- розподіл коштів між нужденними за їх замовленнями та характеристиками;
- відстеження руху коштів бюджету благодійного фонду;
- резервне збереження достовірної інформації фонду з наступним її відтворенням;
- автоматичне генерування звітів щодо отримання та використання коштів;
- формування статистичної звітності для аналізу діяльності фонду;
- підтримка прийняття відповідних рішень.

Отже, благодійний фонд матиме у своєму розпорядженні надійну інформаційну систему [10], що забезпечить автоматизоване та чітке управління його коштами з документуванням усіх проведених операцій, необхідних для формування відповідних звітів, а також формування статистичних даних, придатних для оцінювання ефективності роботи фонду та підготовки відповідних управлінських рішень.

Інструменти розроблення програмного засобу. Під час проектування інформаційної системи було використано такі інструменти:

- *Microsoft Visio 2019* – для побудови UML-діаграм [37];
- *Figma* – векторний сервіс, призначений для прототипування та розроблення інтерфейсу користувача [31];
- *Oracle SQL Developer Data Modeler* – призначений для створення логічної моделі бази даних програмного засобу;
- *Draw.io* – онлайн сервіс, призначений для побудови діаграми розгортання та ER-моделі (від англ. *Entity-Relationship model*, модель "сутність – зв'язок") – модель даних, що дає змогу описувати концептуальні схеми предметної області [42].

На етапі розроблення програмного засобу було використано такі інструменти:

- *Microsoft Visual Studio Community 2019* – середовище, призначене для розроблення ASP.NET MVC 5 веб-додатку [29];
- *SQL Server Management Studio 2018* – утиліта, призначена для підготовки конфігурації, оброблення та виконання компонент СУБД Microsoft SQL Server.

Технології розроблення програмного засобу:

- *ASP.NET MVC 5* – фреймворк, призначений для реалізації веб-засобу з використанням шаблону Model-View-Controller [38, 39];
- *СУБД SQL Server 2017* – система управління базами даних. Як сервер даних виконує головну функцію зі збереження та надання даних у відповідь на запити інших додатків, які можуть виконуватися як на тому ж самому сервері, так і у мережі;
- *Entity Framework 6* – структура об'єктно-реляційного відображення (ORM) з відкритим кодом для ADO.NET. Вона була частиною .NET Framework, але з версії 6 Entity Framework стала відокремленою від .NET Framework [41];
- *Liqpay APIs* – для надання системі доступу до банківського рахунку, створення платежів і отримання архіву платежів.

Отже, проаналізовано можливі інструменти, які можна використовувати для побудови веб-додатку, визначено етапи його життєвого циклу, на яких доцільно використовувати відповідні інструменти і призначення кожного з них [15]. Також розроблено специфікацію вимог до майбутнього програмного засобу, що є основою для розроблення веб-додатку, позаяк містить: функціональні та не функціональні вимоги, вимоги до інтерфейсів користувача та характеристики системи загалом. Визначено основні класи користувачів веб-додатку та встановлено функціональність системи для кожного з них.

Проектування архітектури програмного засобу.

Як уже згадувалося раніше, інформаційну систему реалізуємо у вигляді веб-додатку. Для цього використа-

ємо архітектурний шаблон *MVC*, який застосовують як для проектування архітектури програмного засобу, так і для його розроблення [39].

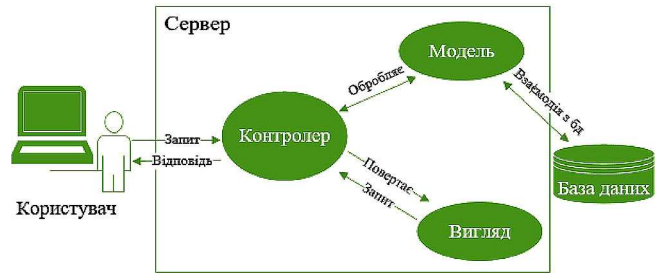


Рис. 2. Архітектура інформаційної системи

Шаблон *MVC* (англ. *Model-View-Controller*) визначає всю архітектуру веб-додатку (рис. 2), передбачає поділ його даних, забезпечує підтримку інтерфейсу користувача та логіку управління даними за допомогою таких компонент [38]:

- *Model* – зберігає дані програмного засобу та управляє ними;
- *View* – відповідає за подання та вигляд інформації. Тобто реалізує інтерфейси користувача з використанням даних, отриманих з моделі; вигляд містить усю необхідну функціональність, з якою може взаємодіяти користувач системи;
- *Controller* – з'єднує Модель та Вигляд для реалізації певної функціональності програмного засобу; контролює та відправляє дані, отримані від користувача з використанням компоненти Вигляд до моделі та навпаки.

Шаблон *MVC* також слугує для відокремлення рівня подання від рівня логіки програми [39]. В шаблоні всі компоненти незалежні одна від іншої, а модифікація кожної з них також не впливає на інші.

Для реалізації веб-додатку було обрано ASP.NET MVC Framework, який використовує шаблон *MVC* та має такі переваги: потужну систему маршрутизації у вигляді простої та зрозумілої адреси URL; легку побудову процесу тестування програмного засобу за рахунок поділу додатку на компоненти *MVC* шаблону; сучасний API-інтерфейс, який забезпечує нові можливості мови програмування та середовища розробки, а саме – ключового слова *await*, лямбда-виразів, *LINQ* (англ. *Language Integrated Query*), а також анонімні та динамічні типи [29].

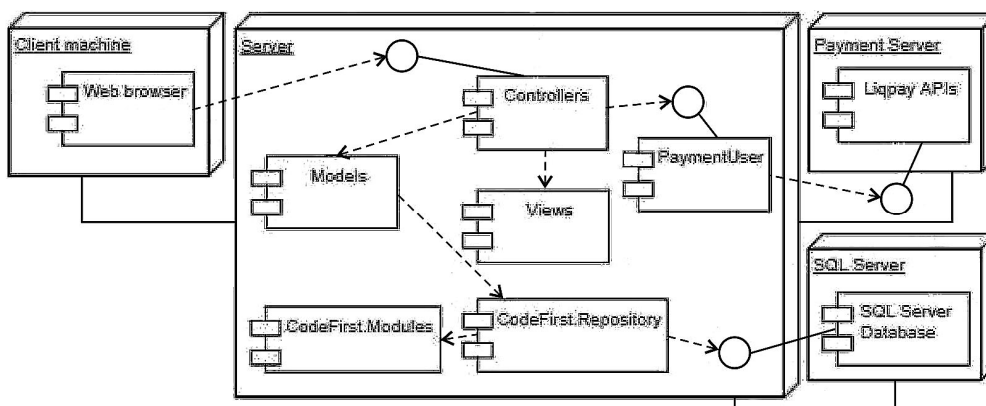


Рис. 3. Діаграма розгортання архітектури веб-додатку

Для наочного подання фізичної та логічної архітектури веб-додатку було побудовано UML-діаграму розгортання (рис. 3) [2]. Вона відображає обчислювальні вузли, які виконують під час роботи програми, а також компоненти, що використовують у цих вузлах [40]. З діаграми видно, що систему буде розміщено на таких вузлах:

- *клієнтська машина* – ЕОМ, за допомогою якої користувач зможе надіслати свої запити до сервера та отримувати від нього відповіді;
- *основний сервер* – призначений для оброблення запитів користувача через звернення до бази даних сервера та платіжного серверу, від яких будуть згенеровані відповіді та відправлені користувачеві;

нент, а також їх взаємозв'язок [13]. З діаграми також видно, що будь-які пожертви чи дії з коштами бюджету фонду може виконувати тільки користувач з правами адміністратора.

У подібний спосіб компоненти взаємодіють між собою і під час використання інших функцій веб-додатку з різними користувачами [24].

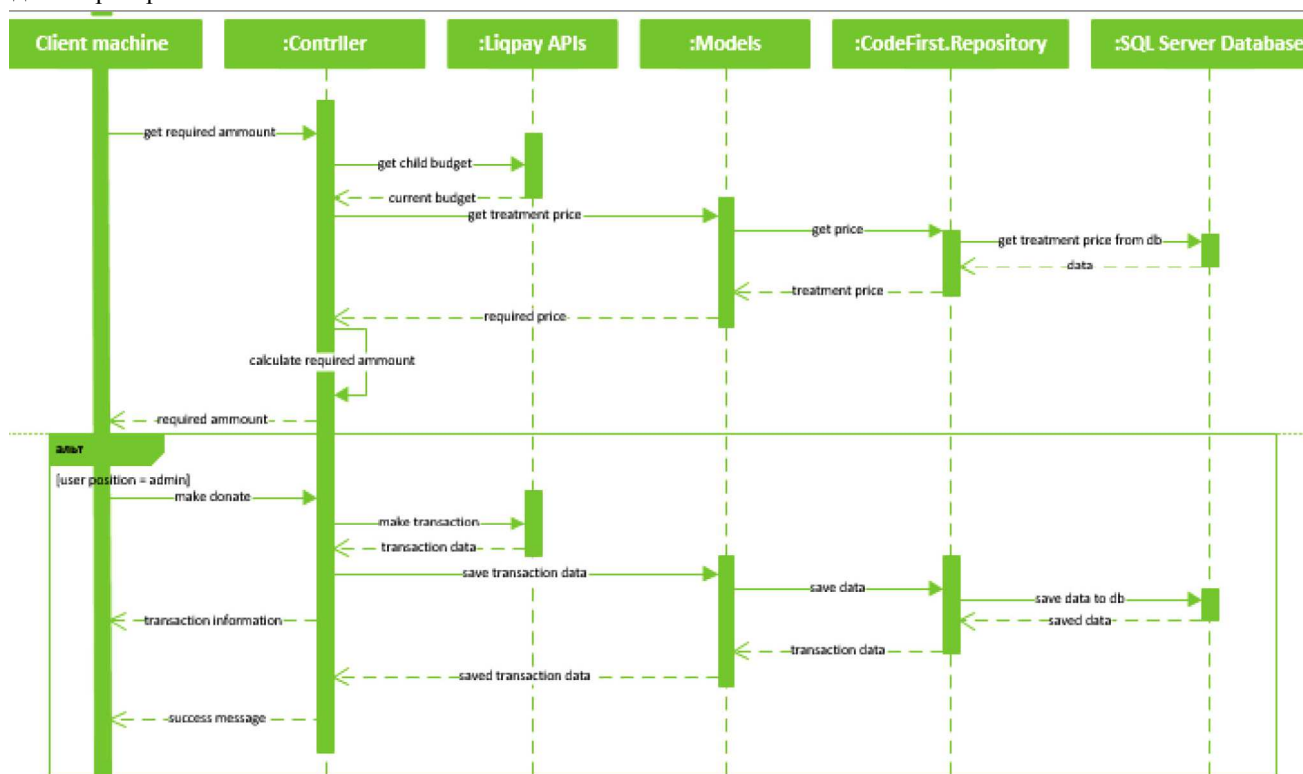


Рис. 5. Діаграма послідовності

Проектування бази даних. Для ефективної та коректної роботи програмного засобу необхідно правильно спроектувати базу даних, що буде містити усю інформацію, потрібну та актуальну для користувача [23]. На підставі цієї інформації він може генерувати відповідні звіти та формувати різні статистики щодо діяльності благодійного фонду. Для проєктованого веб-додатку достатньо однієї реляційної бази даних SQL Server, яка міститиме усі необхідні для роботи таблиці та зв'язки між ними [26]. Оскільки ця база даних важлива складова інформаційної системи, то проектування її моделі будемо виконувати в три етапи – проектування бази даних на концептуальному, логічному та на фізичному рівнях [3].

Концептуальна модель бази даних SQL Server – інформаційна модель найбільш високого рівня абстракції. Як засіб для її побудови було обрано *ER-модель* (англ. *Entity-Relationship model*) з використанням узагальнених конструкцій блоків. Отже, було побудовано модель *сутність-зв'язок*, що описує та визначає відповідну предметну область, а саме – роботу благодійного фонду. Для *ER-моделі* було обрано нотацію "вороняча лапка", оскільки за допомогою неї можна побудувати компактну концептуальну модель бази даних, що містить такі елементи: сутність, що визначає певний об'єкт (позначено прямокутником), ім'я сутності (визначене іменником); зв'язок, що пов'язує дві сутності (позначене лінією); множинність зв'язку (зображено символами); атрибути, що визначають властивості сутностей (записують всередині прямокутника) [42].

Побудову концептуальної моделі (рис. 6) реалізовано за допомогою онлайн інструменту *Lucidchart*, в якій було визначено основні сутності з її головними атрибу-

тами для бази даних програмного засобу, а також зв'язки між ними.

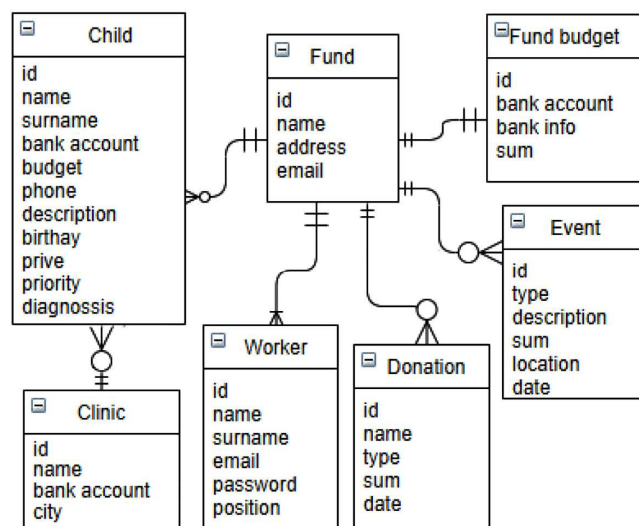


Рис. 6. Концептуальна модель бази даних SQL Server

Логічна модель бази даних SQL Server, робота якої базується на підставі концептуальної моделі. Тут визначено структуру даних у вигляді реляційних таблиць та колонок. Для кожної сутності (таблиці) було описано її атрибути з назвами колонок та відповідними типами даних, а також визначено зовнішні ключі для отримання відповідних відношень між сутностями. Для позначення ідентифікаторів таблиць та їх атрибутів було використано назви, наближені до бізнес-назв предметної області. Для подання логічної моделі (рис. 7) було побудовано реляційну базу даних з використання інструменту *Oracle SQL Developer Data Modeler*.

Фізична модель бази даних SQL Server передбачає потребу визначення, для якої СУБД розробляють базу даних, оскільки це впливає на спосіб її подання та зберігання даних у пам'яті комп'ютера. Також тут потрібно забезпечити цілісність даних шляхом отримання набору

правил, що встановлюють допустимість значень для даних. Окрім цього, було визначено обмеження для первинних та зовнішніх ключів. Як результат, було побудовано фізичну модель бази даних SQL Server у вигляді таблиць та зв'язків між ними (рис. 8).

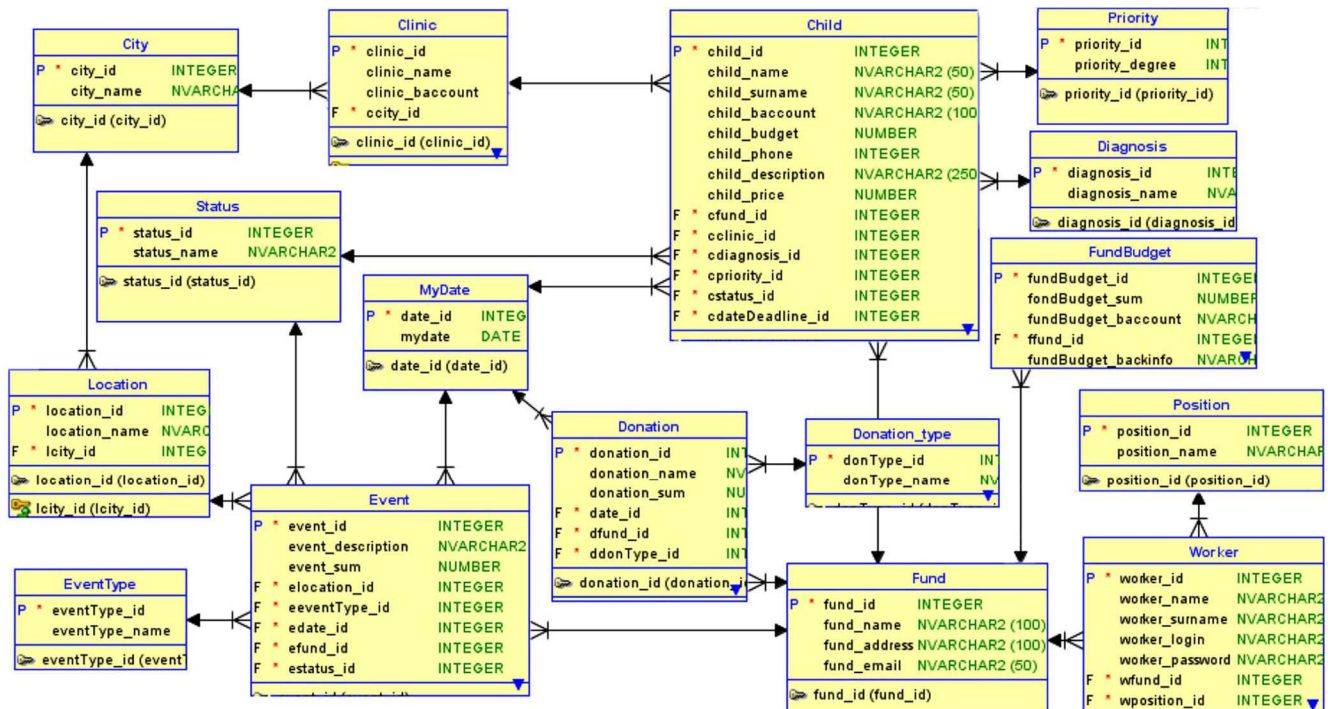


Рис. 7. Логічна модель бази даних SQL Server

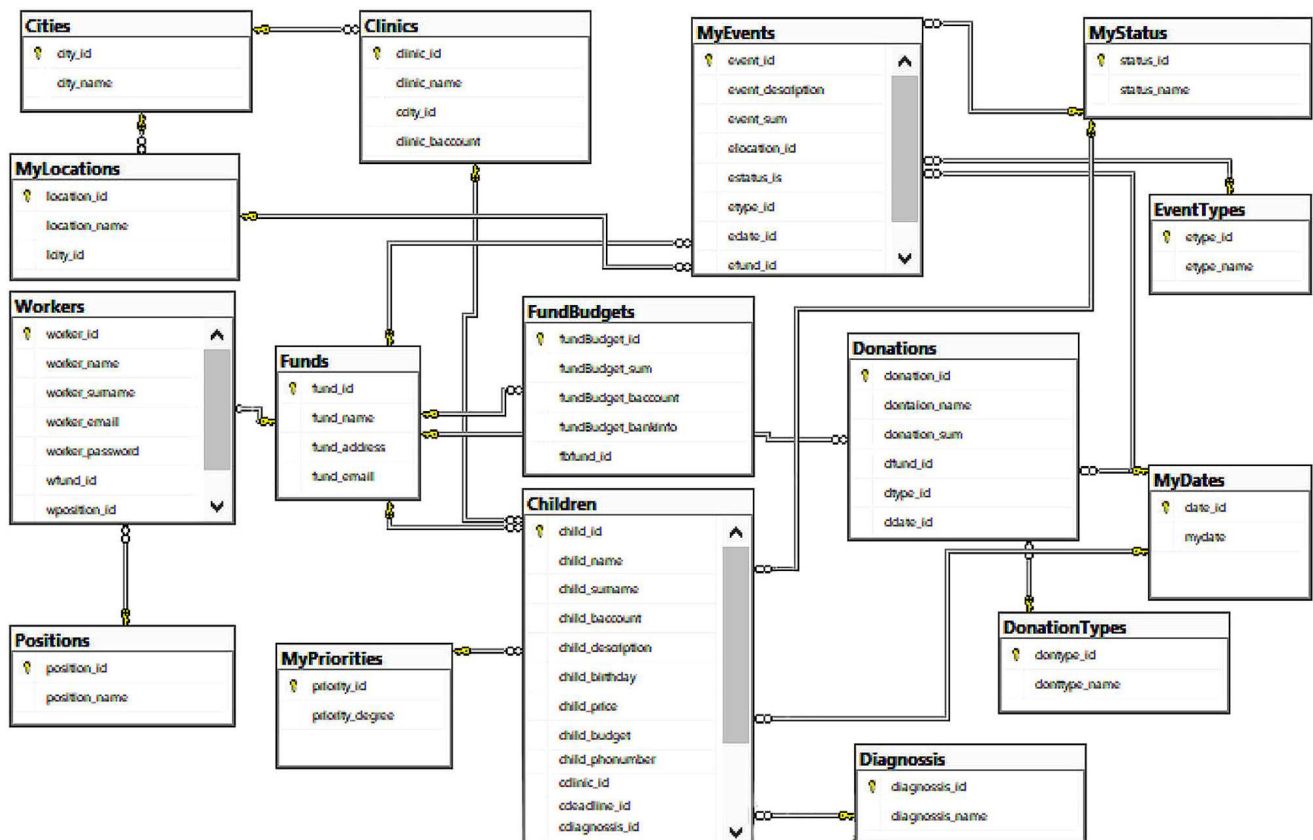


Рис. 8. Фізична модель бази даних SQL Server

Створення прототипу інтерфейсу користувача. Згідно з вимогами до веб-додатку та двох класів користувачів, що матимуть доступ до системи, необхідно спроектувати простий та зручний інтерфейс користувача, за допомогою якого кожен з них матиме змогу вико-

ристовувати функції веб-додатку [1]. Тут необхідно, щоб інтерфейс був зручним та практичним у використанні, оскільки це визначає безпосередню якість роботи програмного засобу [15]. Для побудови правильного інтерфейсу користувача було використано такі принципи [5]:

- *близькість до реального світу* – передбачає використання термінів, близьких до предметної області;
- *управління свободою дій користувача* – використовують для надання користувачеві можливості зайти на будь-яку сторінку веб-додатку;
- *цілісність та стандарти* – забезпечує правильне розташування кнопок відповідно до їх смислового навантаження;
- *запобігання помилкам* – використовують для визначення елементів, які має ввести користувач, які повідомлять його про некоректність введених даних або нададуть поля, наприклад, календар, що унеможливить неправильне введення даних;
- *розпізнавання, а не згадування* – використовують для відображення певних функцій системи, наприклад, редагування або видалення, де було використано відповідні зображення;
- *естетичний та мінімально необхідний дизайн* – визначає, наскільки ефективно користувач зможе використовувати програмний засіб. Проблема в тому, що вікна інтерфейсів можуть мати елементи, що не відповідають функціям системи, або міститимуть інформацію, що не стосується предметної області, тому користувачам-початківцям часто важко зорієнтуватися на підставному призначенні веб-додатку.

Для створення прототипу інтерфейсу користувача було використано онлайн сервіс *Figma* та спроектовано такі основні його елементи [19]:

- *сторінка авторизації* – містить поля для введення логіну та паролю, що здійснює ідентифікацію користувача;
- *бокове та верхнє меню* – фіксовані елементи інтерфейсу. Наприклад, у боковому меню буде розташовано посилання для переходу на такі сторінки, як особи (діти), лікарні, бю-

джет, події, статистика, звіти, працівники та налаштування. Верхнє меню застосовують для переходу на головну сторінку веб-додатку та виходу авторизованого користувача з системи;

- *сторінка дійових осіб* – містить перелік усіх зареєстрованих у системі осіб (дітей) з можливістю додавання нових;
- *сторінка додавання нових осіб* – містить усі необхідні поля для їх заповнення користувачем, щоб зареєструвати особу в системі;
- *сторінка подій* – містить перелік заходів, організованих фондом з можливістю створення нових;
- *сторінка створення події* – містить необхідні поля, які потрібно заповнити для створення нового заходу;
- *сторінка бюджету* – містить актуальну інформацію про кошти, що зберігаються на рахунку фонду чи на рахунках його проєктів, реквізити та актуальну інформацію про зміст рахунку, а також випадний перелік з подіями та полями, призначеними для редагування;
- *сторінка закладів* – містить інформацію про клініки/лікарні, де особам будуть надавати безпосередню допомогу, а також поля для додавання нових.

Також було визначено основний шаблон інтерфейсу, використання якого забезпечить розроблення всіх сторінок веб-додатку, зокрема: сформовано основну гамму кольорів, визначено основні геометричні фігури та їх розташування на сторінці тощо [20, 27]. Приклади спроектованих елементів графічного інтерфейсу користувача зображено на рис. 9.

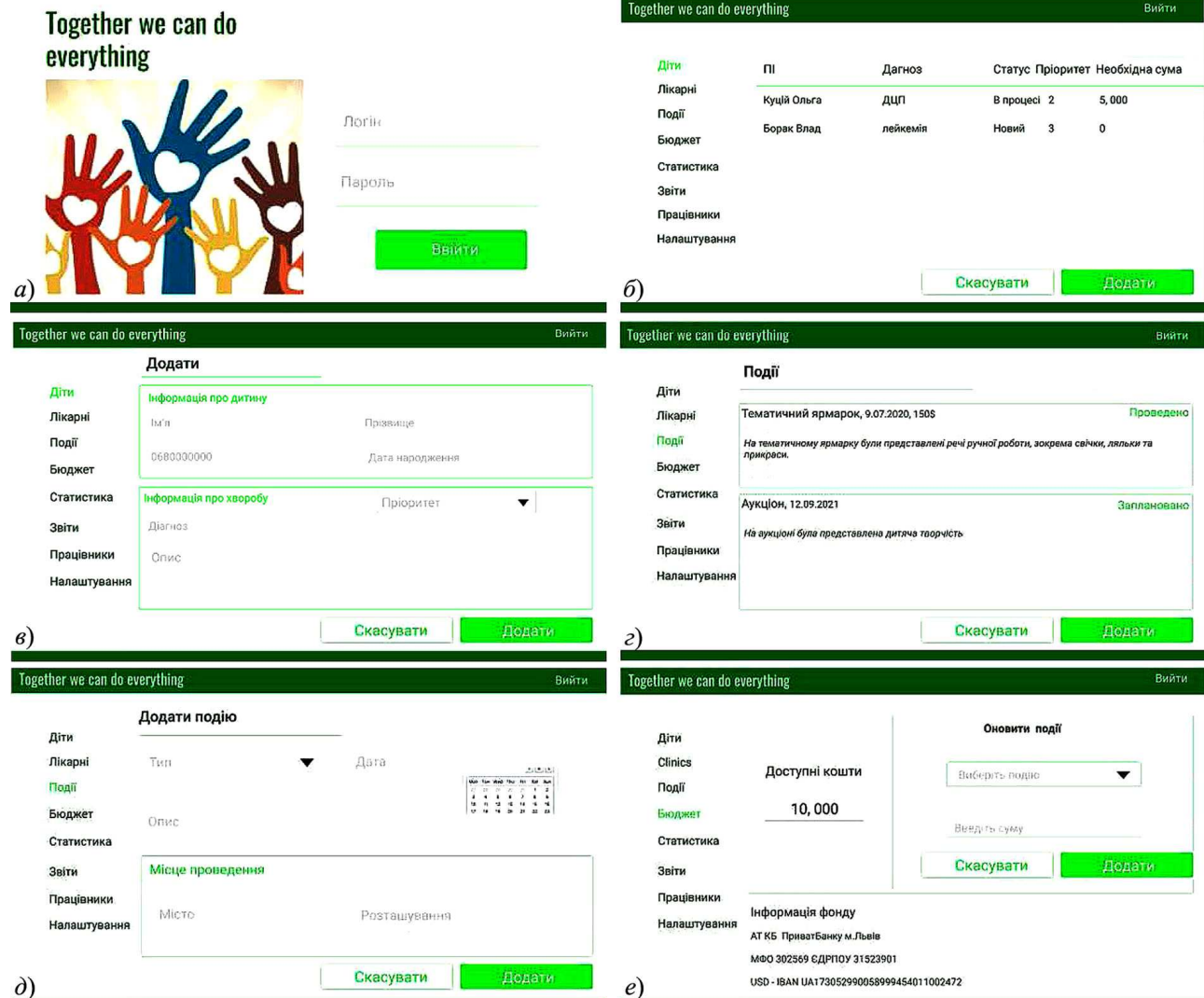


Рис. 9. Прототипи сторінок веб-сайту: а) авторизації, б) з переліком осіб (дітей), в) додавання нової особи в систему, г) події, д) створення нової події, е) бюджету фонду

Обговорення результатів дослідження. Інформаційна система для організації благоїного процесу має містити методи і способи збору, нагромадження, реєстрації, передавання, відображення, зберігання, пошуку, модифікації, аналізу, захисту та видачі необхідної інформації всім зацікавленим сторонам за допомогою відповідних апаратних і програмних засобів [4 с. 11; 25, с. 117; 26].

Веб-сайти за способом оброблення такої інформації мають містити забезпечувальні та функціональні складові [4, с. 12; 17, 35]. Зазвичай, забезпечувальна складова сайту призначена для вирішення завдань обліку, контролю та аналізу інформації. Містить технічне, інформаційне, технологічне, математичне, організаційне, правове, економічне та інші види забезпечення [8, 9, 11]. Водночас, функціональна частина веб-сайту – це модель системи управління об'єктом (благоїним фондом), яка синтезує забезпечувальну та предметну технології, що реалізують функціонування системи управління за певними правилами [3, с. 13; 11, 23, 25, с. 117]. Предметна технологія передбачає послідовність етапів, що мають перетворити первинну інформації на очікуваний результат у предметній галузі – облік ресурсів і їх розподіл [10, 16]. Сукупність забезпечувальних і функціональних складових веб-сайту реалізують робочі місця бухгалтера та інших працівників, які є основою облікового процесу [3, с. 3–8]. Тому складова веб-сайту, яка здійснює управління коштами благоїного фонду – це не що інше, як бухгалтерська система, яка й забезпечує таке управління [17].

З огляду на зазначене вище, систему управління коштами благоїного фонду можна подати як організаційну структуру, у яку входить набір робочих місць, що чітко взаємодіють між собою при здійсненні облікових функцій [10, 24]. До цієї структури належать робочі місця бухгалтера, а також працівників, більш чи менш пов'язаних з організацією бухгалтерського обліку – з документування, реєстрації та оформлення різних документів і звітів тощо [23].

Відповідно до місця організації та технології виконання функціональних завдань веб-сайту, що передбачає насамперед бухгалтерський облік коштів благоїного фонду, можна поділити на три групи [16, 25, с. 117]:

- 1) підготовка інформації та її оброблення в бухгалтерії;
- 2) систематизація й узагальнення облікової інформації на рахунках за видами ресурсів, а також контроль і аналіз результатів систематизації та узагальнення;
- 3) формування інформації для подальшого використання та її координація.

Ретельно налагоджений бухгалтерський облік благоїного фонду має своєчасно забезпечувати управлінські потреби зацікавлених сторін необхідною та достовірною інформацією [4]. Окрім цього, надійний та прозорий облік благоїних коштів є основою інформаційною базою для всебічного аналізу фінансово-господарської діяльності благоїного фонду, обґрунтування відповідних управлінських рішень щодо витрат бюджету фонду та контролю громадськості за їх цілеспрямованість та достовірність.

Водночас, можливість використовувати інформаційні технології в процесі управління коштами благоїного фонду робить його більш конкурентоздатним, зокрема за рахунок підвищення оперативності щодо аналізу

результатів діяльності та прийняття обґрунтованих управлінських рішень, керованості й адаптованості до змін ринкової кон'юнктури тощо.

Висновки

Спроектвано систему управління коштами благоїного фонду, яка дасть змогу сформувати взаємодію учасників процесу в окремій інформаційній системі задля забезпечення їх актуальною, достовірною та цілісною інформацією на етапах отримання коштів, розподілу та аналізу використання. За результатами виконання роботи можна зробити такі основні висновки.

1. З'ясовано, що таке благоїність та її актуальність на сьогодні. Описано основні способи, використовуючи які кожна небайдужа людина може взяти участь для вирішення певного виду проблем нужденних осіб. Встановлено, що для ефективної роботи фонду потрібен бюджет, який у майбутньому буде розподілено між нужденними. Перераховано основні джерела постачання коштів для збільшення бюджету фонду та описано можливий спосіб розподілу коштів між нужденними особами. Наведено фінансову інформацію, що використовують для роботи благоїного фонду та етапи її оброблення з подальшим документуванням.

2. Проаналізовано сучасні програмні засоби, що використовують для роботи благоїних фондів. Описано найпопулярніші сайти та платформи України, наведено їх основні функції, а також правила роботи благоїних фондів, які потрібно врахувати в майбутньому програмному засобі для ефективної, правильної та якісної його роботи. У такий спосіб робота благоїного фонду буде максимально автоматизованою, що дасть йому можливість ефективно вирішувати певні проблеми нужденних осіб, стане практичною для проведення необхідних грошових операцій та інформаційно актуальною для проведення аналізу та формування відповідних висновків.

3. Визначено основну мету та головні цілі розроблення програмного засобу, що допомогло краще зрозуміти функціональність системи управління коштами благоїного фонду для отримання очікуваного результату. Проаналізовано можливі інструменти, які можна використовувати для створення веб-додатку, визначено етапи його життєвого циклу, де буде доцільно використовувати відповідні інструменти і призначення кожного з них зокрема.

4. Розроблено специфікацію вимог до програмного засобу, що є основою для розроблення веб-додатку, позаяк містить: функціональні та не функціональні вимоги, вимоги до інтерфейсів користувача та характеристики системи загалом. Визначено основні класи користувачів веб-додатку та встановлено функціональність системи для кожного з них.

5. Визначено основне архітектурне рішення, яке використано для побудови веб-додатку. З'ясовано, що додаток буде реалізовано на підставі архітектурного шаблону *MVC*, який застосовують як для проектування архітектури програмного засобу, так і для його розроблення. Визначено основні компоненти, необхідні для повноцінної роботи веб-додатку, а також описано технологію, яку буде використано для його реалізації на підставі визначеної архітектури.

6. Згідно з визначеними функціями та характеристиками веб-додатку, класами користувачів і їх доступнос-

ті до функціональності системи, спроектовано поведінку системи з використанням UML-діаграм. Насамперед було побудовано діаграму варіантів використання, що показує усіх користувачів системи, а також ту функціональність, доступ до якої має відповідний клас користувачів. Для відображення динамічної роботи веб-додатку було побудовано діаграму послідовності, яка відтворює процес виділення коштів з бюджету фонду для лікування нужденних осіб, описуючи компоненти системи, послідовність їх використання та взаємозв'язки між ними.

7. Спроектовано базу даних веб-додатку, процедура якого передбачає виконання таких етапів: проектування на концептуальному рівні з використання ER-моделі; проектування на логічному рівні, внаслідок якої було побудовано реляційну базу даних; проектування на фізичному рівні в рамках відповідної СУБД, для якої було розроблено відповідну базу даних.

8. Спроектовано основний прототип графічного інтерфейсу користувача, а саме: головні його елементи згідно з відомими принципами їх побудови; визначено загальну структуру та основне сприйняття його вигляду. На підставі запропонованих рішень розроблено усі інші компоненти, необхідні для роботи веб-додатку, графічні елементи та сторінки.

References

- Alekseev, A. (2019). *Introduction to Web Design*. Tutorial. Moscow: DMK Press, 184 p. [In Russian].
- Arlow, D., & Neustadt, I. (2007). *UML 2 and Unified Process. Practical object-oriented analysis and design*. (2nd ed.). (Trans. From English). St. Petersburg: Symbol-Plus, 624 p.
- Belukha, M. T. (2003). Application of the accountants workstation in accounting and control at the enterprise. *Accounting and Auditing*, 12, 3–8. [In Ukrainian].
- Benko, M. M. (2006). *Information systems and technologies in accounting*. Tutorial. For university students. Kyiv: National University of Trade and Economics, 362 p. [In Ukrainian].
- Clifton, Jan. (2017). *Designing the user interface in Android*. Moscow: DMK Press, 452 p. [In Russian].
- Dikov, A. V. (2019). *Client-side technologies of web design. HTML5 and CSS3*. Tutorial. Moscow: Doe, 188 p. [In Russian].
- Duquette, D. (2019). *HTML and CSS. Website development and design*. Moscow: Eksmo, 480 p. [In Russian].
- Garrett, J. (2015). *Web design. Elements of interaction experience*. St. Petersburg: Plus sign, 192 p. [In Russian].
- Garrett, Jess. (2020). *Web design. Elements of interaction experience*. Moscow: Symbol-Plus, 285 p. [In Russian].
- Guzhva, V. M. (2005). *Information systems and technologies in enterprises*. Tutorial. Kyiv: KNEU, 400 p. [In Ukrainian].
- Kirsanov, D. (2015). *Web design: a book by Dmitry Kirsanov*. Moscow: Symbol, 368 p. [In Russian].
- Kiselev, S. V. (2019). *Web design*. Moscow: Academia, 285 p. [In Russian].
- Kohler, Adrian, & Bradsky, Gary. (2017). *Studying OpenCV 3*. Moscow: DMK Press, 826 p. [In Russian].
- Law of Ukraine of July 5, 2012 (2013). *About charitable activity and charitable organizations*. Information of the Verkhovna Rada of Ukraine, № 25, Art. 252. [In Ukrainian].
- Levus, E. V., Marusenkov, T. A., & Nitrebich, O. O. (2017). *Software life cycle*. Lviv: Lviv Polytechnic Publishing House, 208 p. [In Ukrainian].
- Luchko, M. R., & Adamik, O. V. (2016). *Information systems and technologies in accounting and auditing*. Tutorial. Ternopil: TNEU, 252 p. [In Ukrainian].
- Matvienko, O. V., & Tsyvin, M. N. (2005). *Fundamentals of information systems management*. Tutorial. Kyiv: Center for Educational Literature, 176 p. [In Ukrainian].
- McFarland, David. (2018). *The new big book on CSS*. Moscow: Peter, 720 p. [In Russian].
- McNeil, P. (2017). *Web design. Web developers book of ideas*. St. Petersburg: Peter, 480 p. [In Russian].
- Minnick, Chris, & Tittel, Ed. (2019). *HTML5 and CSS3 for Dummies*. Moscow: Dialectics, 400 p. [In Russian].
- Nielsen, J. (2015). *Web Design: A Book by Jacob Nielsen*. Moscow: Symbol, 512 p. [In Russian].
- Okhrimenko, V. M., & Voronkova, T. B. (2006). *Information systems and technologies at enterprises*. Lecture notes (for students and listeners of FPO and ZN "Economics of Enterprise"). Kharkiv: KNAMG, 185 p. [In Ukrainian].
- Pavlysh, V. A., Glinenko, L. K., & Shakhovska, N. B. (2018). *Fundamentals of information technology and systems: a textbook*. Lviv: Lviv Polytechnic Publishing House, 620 p. [In Ukrainian].
- Petrochenkov, A., & Novikov, E. (2017). *Perfect Landing Page. We create selling web pages*. St. Petersburg: Peter, 320 p. [In Russian].
- Sopko, V. V., & Zavgorodnyy, V. P. (2004). *Organization of accounting, economic control and analysis: a textbook*. For university students. Kyiv: KNEU, 412 p. [In Ukrainian].
- Susidenko, V. T. (2016). *Information systems and technologies in accounting and auditing*. Tutorial. Kyiv: Center for Educational Literature, 224 p. [In Ukrainian].
- Syrykh, Yu. A. (2019). *Modern web design. Desktop and mobile*. Moscow: Dialectics, 384 p. [In Russian].
- Website. (2021). "Thank you" for breakfast, "thank you" for dinner, or for what charitable foundations live on. Retrieved from: <https://life.pravda.com.ua/columns/2020/01/28/239718/>. [In Ukrainian].
- Website. (2021). *Benefits of ASP.NET MVC*. Retrieved from: https://professorweb.ru/my/ASP_NET/mvc/level1/1_2.php. [In Ukrainian].
- Website. (2021). *Charity is the greatest manifestation of humanity*. Retrieved from: <https://archangel-of-light.org.ua/blagodiy-nist-naybilshiy-viyav-gumannosti>. [In Ukrainian].
- Website. (2021). *EF 6 Code-First*. Retrieved from: <https://www.entityframeworktutorial.net/code-first/what-is-code-first.aspx>.
- Website. (2021). *Eight fundraising ideas for non-profit organizations: foreign experience*. Retrieved from: <https://eupros-tir.org.ua/practices/144495>. [In Ukrainian].
- Website. (2021). *Everyone Can Do International Charitable Foundation*. Retrieved from: <https://www.everybodycan.com.ua/>. [In Ukrainian].
- Website. (2021). *How to accept and provide assistance to a charitable foundation*. Retrieved from: [https://www.pros-tir.ua/?blogs=yak-virno-blahodijnomu-fondu-pryjmaty-ta-nadavaty-dopomohu\(2020\)](https://www.pros-tir.ua/?blogs=yak-virno-blahodijnomu-fondu-pryjmaty-ta-nadavaty-dopomohu(2020)). [In Ukrainian].
- Website. (2021). *Information system*. Retrieved from: https://uk.wikipedia.org/wiki/Information_system. [In Ukrainian].
- Website. (2021). *Lets help International Charitable Foundation*. Retrieved from: <https://lets-help.com.ua/>. [In Ukrainian].
- Website. (2021). *Microsoft Visual Studio development environment*. Retrieved from: https://informatics.in.ua/programming_csharp/part_01.php. [In Ukrainian].
- Website. (2021). *Model view controller*. Retrieved from: <https://ru.wikipedia.org/wiki/Model-View-Controller>.
- Website. (2021). *MVC Architecture*. Retrieved from: <https://www.educative.io/blog/mvc-tutorial>.
- Website. (2021). *Sequence diagram*. Retrieved from: <http://flash.retejo.info/cxefpagxo/uml/diagrama-poslidovnosti>. [In Ukrainian].
- Website. (2021). *Syncfusion DocToPDFConverter for.NET Framework*. Retrieved from: <https://nugetmusthaves.com/Package/Syncfusion.DocToPdfConverter.Base>.
- Website. (2021). *The "essence-connection" model*. Retrieved from: https://en.wikipedia.org/wiki/Model_essence_-_relationships. [In Ukrainian].
- Website. (2021). *Ukrainian charity exchange dobro.ua*. Retrieved from: <https://dobro.ua/>. [In Ukrainian].
- Website. (2021). *What is charity?*. Retrieved from: <https://en.wikipedia.org/wiki/Charity>. [In Ukrainian].

DESIGNING A FUND MANAGEMENT SYSTEM FOR A CHARITY FUND

A charitable fund management system has been designed. The system will ensure the interaction of process participants in a separate information system to provide them with relevant, reliable and complete information at the stages of receiving funds, their distribution and analysis of use. It was found that the relevant information system for the organization of the charity process should include methods and ways of collecting, accumulating, registering, transmitting, displaying, storing, searching, modifying, analyzing, protecting and disseminating the necessary information to all stakeholders. The most popular sites and platforms of Ukraine for the work of charitable foundations are analyzed, which helped to better understand the functionality of the organization's funds management system and their main purpose. The main architectural solution is defined – the MVC template, which was used to design and develop the software architecture. The main components necessary for the full operation of the web application are identified, as well as the technology that will be used for its implementation on the basis of a certain architecture.

According to the defined functions and characteristics of the web application, user classes and their availability to the system functionality, the system behavior is designed using UML-diagrams. The diagram of variants of use of system of management of means is constructed. It shows all users of the system, as well as the functionality of the system, which has access to the appropriate class of users. A sequence diagram was built to show the dynamic operation of the web application. It reproduces the process of allocating funds from the fund's budget for the treatment of needy people. The database of the web application is designed, which provides for the following stages: design at the conceptual level using the ER-model; design at the logical level, as a result of which a relational database was built; design at the physical level within the relevant DBMS for which the relevant database was developed. Developed a graphical user interface: its main elements in accordance with the known principles of their construction; the general structure and the basic perception of its kind are defined. Based on the proposed solutions, all other components necessary for the operation of the web application, graphics and pages have been developed.

Keywords: information system; web application; architectural solution; use case diagram; sequence diagram; relational database.

Ознайомитись з вимогами до авторів статей можна на сайті видання:

<https://nv.nltu.edu.ua>

Підп. до друку 30.04.21. Формат 60×84/8. Папір офсетний. Друк офсетний.
Ум. др. арк. 15,35. Ум. фарбо-відб. 15,81. Облік.-вид-арк. 15,58. Тираж 250 прим.
Зам. № 28/2021

Видавець: Редакційно-видавничий центр НЛТУ України
79057, м. Львів, вул. Генерала Чупринки, 103
Тел.: (032) 240-23-50
Email: nv@nltu.edu.ua <https://nv.nltu.edu.ua>

Свідоцтво про внесення суб'єкта видавничої справи до
Державного реєстру видавців, виготовників і розповсюджувачів видавничої продукції
(Серія ДК, № 2062 від 17.01.2005 р.)

Свідоцтво про державну реєстрацію друкованого засобу масової інформації
(Серія КВ, № 11889-760ПР від 26.10.2006 р.)

Згідно з наказом Міністерства освіти і науки України від 17.03.2020 р., № 409,
"Науковий вісник НЛТУ України" належить до Переліку наукових фахових видань України (категорія Б),
в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора
і кандидата наук за галузями науки – біологічні, сільськогосподарські, технічні та економічні
й такими спеціальностями: 051 – економіка; 091 – біологія; 101 – екологія; 187 – деревообробні та меблеві техно-
логії; 205 – лісове господарство; 206 – садово-паркове господарство, а наказом від 2.07.2020 р., № 886 – такими
спеціальностями: 122 – комп'ютерні науки; 126 – інформаційні системи та технології.