

Коцун В.І.

ПВНЗ «Європейський університет»

ВЕБ-СЕРВІС ПОШУКУ ЗНАЙДЕНИХ ТА ЗАГУБЛЕНИХ РЕЧЕЙ «БЮРО ЗНАХІДОК»

У статті розглядається реалізація веб-сервісу пошуку знайдених та загублених речей. Веб-сервіс розроблено використовуючи скриптову мову програмування PHP, фреймворк Symfony, мову JavaScript та її бібліотеку jQuery, технологію Ajax, а також розмітки HTML та таблиць стилів CSS.

В сучасному соціумі важко уявити собі людину без стандартного набору персональних речей, таких як ключі, телефон, гаманець, кишенькові гроші тощо, але в шаленому ритмі життя є велика вірогідність загубити важливий предмет. В більшості випадків невеликі, важливі предмети, такі як ключі, телефон, паспорт, гаманець та інші, знаходяться, але все ж таки буває, що доводиться використовувати допомогу різних сервісів, служб для допомоги в пошуку важливих речей. До таких служб відносяться бюро знахідок, дошки оголошень, ломбарди, сповіщення по радіо, телебаченню тощо.

Проаналізовано існуючі сервіси пошуку знайдених і загублених речей та показано їх основні переваги і недоліки. Наведено основні фактори необхідності реалізації веб-сервісу пошуку знайдених та загублених речей. Обґрунтовано доцільність використання мови програмування PHP, як однієї з найпоширеніших мов, що використовується у сфері веб-розробки, для розробки системи, яка має відповідати всім поставленим вимогам.

У роботі реалізована можливість створення заявки про знайдену або загублену річ, перегляду вже існуючих заявок, зв'язку з користувачами, які подали заявки, авторизації через соціальну мережу Facebook, модерування заявок, використання карти для вказування місця знайденого, або загубленого предмету, отримання статистики стосовно створених заявок.

Показано, що завданням створеної системи є зменшення навантаження на органи муніципальної влади у роботі з предметами, які надходять до бюро знахідок, та створення веб-сервісу, який би був універсальний для всього світу.

Ключові слова: веб-сервіс, бюро знахідок, програмування, фреймворк, JavaScript.

Постановка проблеми. В сучасному соціумі важко уявити собі людину без стандартного набору персональних речей, таких як ключі, телефон, гаманець, кишенькові гроші тощо, але в шаленому ритмі життя є велика вірогідність загубити важливий предмет. Для того, щоб зменшити вірогідність втрати таких речей, людина винайшла величезну кількість допоміжних заходів, наприклад, для телефону – кобуру на пояс, для ключів – ланцюжок до ремня штанів і так далі, але в результаті речі все-одно губляться через необережність чи через погану пам'ять. Це здебільшого невеликі, важливі предмети, такі як ключі, телефон, паспорт, гаманець та інші. Переважно вони знаходяться, але все ж буває, що доводиться використовувати допомогу різних сервісів, служб для допомоги в пошуку важливих речей. До таких служб відносяться бюро знахідок, дошки оголошень, ломбарди, сповіщення по радіо, телебаченню тощо.

Середньостатистично власники загублених речей звертаються тільки за 15% всіх знахідок, тобто про 85% речей люди не пам'ятають або не

хочуть їх забирати, тому виникає потреба в розробці сервісу, який допоможе людям знаходити свої загублені речі максимально зручно і просто.

Аналіз останніх досліджень і публікацій.

У світі існує величезна кількість сервісів та їх аналогів, які успішно використовуються для пошуку речей. Мабуть, у кожній країні є бюро знахідок і більшість з них підпорядковуються державній структурі. Також існують різні служби, які закріплені за місцями з великим скупченням людей: аеропортами, готелями, автостоянками, залізничними вокзалами та ін.

Одним із найпопулярніших сервісів зараз є Lost And Found.com. До його переваг можна віднести досить велику наповнену базу знайдених та загублених речей, велику кількість користувачів, функціонал для багатьох країн світу. Проте, він має ряд недоліків: старий та незручний інтерфейс, значна кількість помилок у роботі, малі шанси на знайдення загубленої речі, дуже повільний, немає соціальної складової, може бути використаний зловмисниками для отримання чужих речей [1].

Нижче наведено інші відомі сервіси, які використовуються для пошуку речей. Сервіс US Berkeley Main Lost And Found надає інформацію про знайдені речі, контактні дані та адресу пункту, де можна їх забрати. Він підпорядковується міліцейському департаменту університету Берклі в штаті Каліфорнія [2]. До переваг можна віднести простоту пошуку втрачених речей, інформаційну безпеку користувачів та постійні оновлення списків речей. Проте недоліками даного сервісу є те, що він працює локально, тобто він доступний лише для студентів університету, якому він підпорядковується, та неінформативний користувацький інтерфейс.

Простий сервіс Airport Lost and Found для пошуку загублених речей у аеропортах світу [3]. Сервіс має значну кількість переваг: приємний інтерфейс, який допомагає взаємодіяти користувачам із сервісом, велику кількість зв'язків із найбільшими аеропортами світу, особистий мобільний додаток для телефонів, які підтримують операційними системами iOS, для кожного предмету існує вичерпна інформація, також користувачі мають змогу використати прийнятну систему фільтрів за різними критеріями для пошуку власних речей. Недоліками сервісу є його вузькоспеціалізованість, а саме спеціалізація виключно на аеропортах. Також відсутня багатомовність на сайті.

ReturnMe онлайн сервіс пропонує власну продукцію для захисту своїх особистих речей – наліпку з ідентифікаційним кодом [4]. Користувач купує наліпку з унікальним кодом та клеїть її на свою річ. Потім, якщо він втрачає свою річ, то той, хто її знайде, може зайти на сайт і по коду, який був вказаний на наклейці, отримати контакти власника.

Сервіс LuckFind від українських розробників, який найбільш схожий за функціоналом до проекту, який пропонується у роботі [5]. Тут присутня можливість створювати заявки про загублені або знайдені предмети, додавати фото та опис і місце на карті. Також присутня реєстрація користувача, і для авторизованого користувача доступний функціонал зв'язку з автором заявки. Сайт є багатомовним та має пошук.

До недоліків сервісу LuckFind можна віднести відсутність авторизації через соціальну мережу Facebook, що значно спростило б цей процес для користувача. Незручний інтерфейс користувача, важко знайти перехід на розділ саме загублених або знайдених речей. Також відсутня можливість вказати на карті зону, де річ могла бути загублена,

адже люди не завжди знають точне місце втрати речі.

Сервіс VARTA1 – дошка оголошень втрачених речей міста Львова [6]. Тут окрім того, що можна додати знайдену або загублену річ, також є можливість повідомити про те, що у вас щось викрали і також присутній розділ «барахолка». Недоліками ж сервісу є незручний інтерфейс, його локальність, відсутня карта з втраченими і знайденими речима, відсутність можливості зв'язку за допомогою сайту, необхідно в описі до речі вказувати свої контактні дані.

Загальним недоліком всіх перерахованих сервісів є відсутність багатомовності, тобто кожен з сервісів відображає інформацію лише на англійській мові без підтримки інших мов. Також недоліком багатьох сервісів є орієнтованість на конкретну країну або інфраструктурну одиницю, та не зручний інтерфейс.

Оскільки кожен з цих сервісів має недоліки, розробка аналогу є актуальною, якщо врахувати недоліки конкурентів та виправити їх.

Постановка завдання. Метою роботи є розробка веб-сервісу, який допоможе людям знаходити свої загублені речі максимально зручно і просто.

Виклад основного матеріалу дослідження. Для створення системи, оновлення та керування інформацією про знайдені та загублені предмети потрібно обрати єдині алгоритми та технології, на базі яких будуватиметься майбутній проект, що забезпечить подільність системи на окремі частини функціоналу, модулі (модуль оголошення, модуль списків і так далі), які зменшать складність подальшої підтримки та оновлення проекту. Тому для розробки системи, яка має відповідати всім поставленим вимогам, вибрано мову програмування PHP – це одна з найпоширеніших мов, що використовується у сфері веб-розробки, підтримується переважною більшістю хостинг-провайдерів та досить низький рівень входження, що забезпечує швидке опанування мови [7].

Однією з основних переваг PHP є наявність інтерфейсів до багатьох баз даних у PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, MongoDB. Для того, щоб система відповідала всім критеріям, в процесі розробки потрібно використати фреймворк – основна концептуальна система або структура для вирішення комплексних задач. Для проекту був вибраний фреймворк Symfony [8]. Symfony спрямований на прискорення створення та підтримки веб-застосунків, а також для уникнення витрат часу для розв'язування

тривіальних задач у розробці (наприклад, написання валідаторів форм).

Для розробки проекту було обрано клієнт-серверну. Візуальне відображення компонент системи (рис. 1).

Для більш наглядного представлення компонентів, їх фізичного знаходження та варіантів комунікації між компонентами було побудовано діаграму розгортання (рис. 2).

Перед початком розробки сервісу потрібно визначитись та розробити структуру бази даних. Це один з найважливіших кроків так як в майбутньому нормально спроектована база даних дасть змогу розширювати та поповнювати функціонал таким чином, щоб сервіс залишався стійким до навантаження.

Тому було спроектовано таку структуру бази даних із таблиць:

- 1) «Користувач» – для зберігання персональних даних користувачів;
- 2) «Категорія» буде містити категорії предметів які загубили або знайшли;
- 3) «Предмет» – для зберігання всієї інформації про речі.

Кожна таблиця має свою специфічну структуру даних, яка дає змогу зберегти інформацію в тому вигляді, в якому вона потрібна розробнику для виконання поставлених завдань. У базі даних використались різні зв'язки, які надали змогу структурно зв'язати дані між собою.

Для створення користувацького інтерфейсу використовується популярний CSS фреймворк Twitter Bootstrap [9] з використанням теми Yeti, в якому реалізовані стилі для розміщення та вигляду стандартних компонентів інтерфейсу. Приклад навігації у шаблоні Yeti зображений на рис. 3.

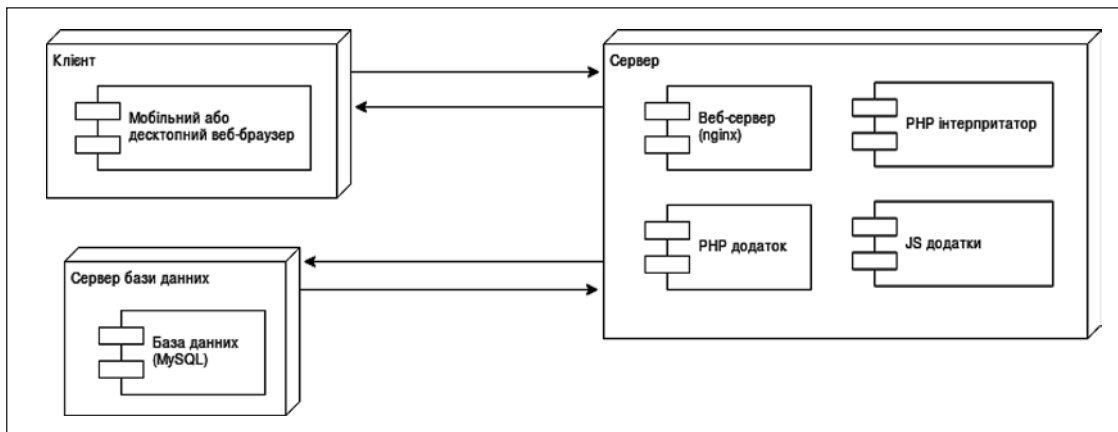


Рис. 1. Діаграма компонент веб-сервісу

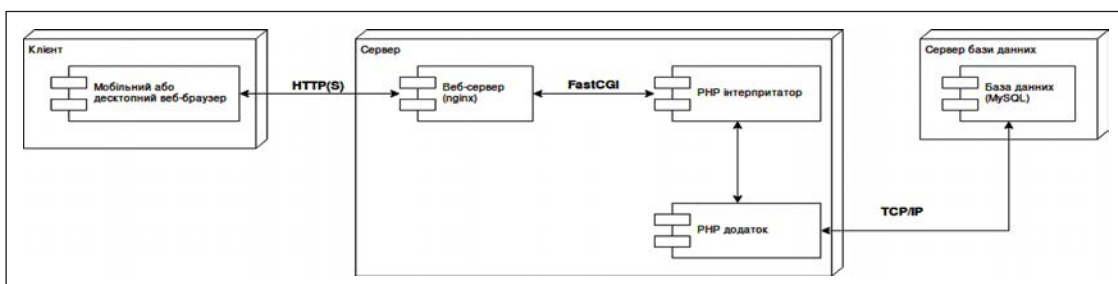


Рис. 2. Діаграма розгортання веб-сервісу



Рис. 3. Приклад навігації у шаблоні Yeti

У результаті кожна сторінка має сформовану панель навігації, яка розміщується вверху сторінки, що дає змогу швидко переключатися між головною та іншими сторінками сервісу, а також швидко авторизуватися, чи зареєструватися користувачу. Представлена на рис. 4.

Відповідно кожна сторінка має свою частину інформації, що розміщується посередині, яка представлена на рис. 5.

Для того, щоб сервіс функціонував ефективно, потрібно запрограмувати алгоритм, основним завдання якого є можливість вибору користувачем будь-якого функціоналу в межах сервісу. Для цього потрібно розробити алгоритми перегляду та додавання заявок про речі, якому передуватиме алгоритм надсилання запитів до бази даних.

Для «спілкування» з базою даних використано класи, які називаються репозиторії. Репозиторії



Рис. 4. Панель навігації сервісу

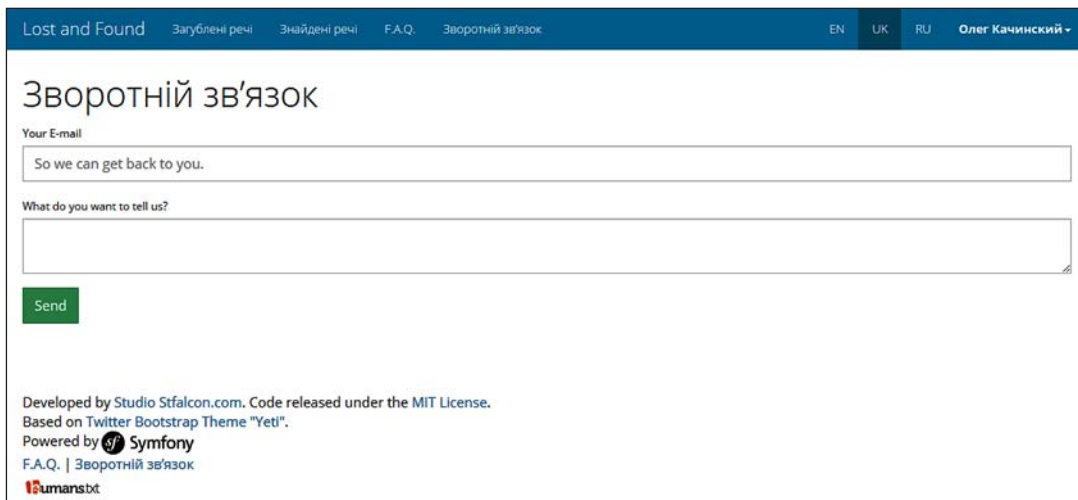


Рис. 5. Приклад сторінки зворотнього зв'язку

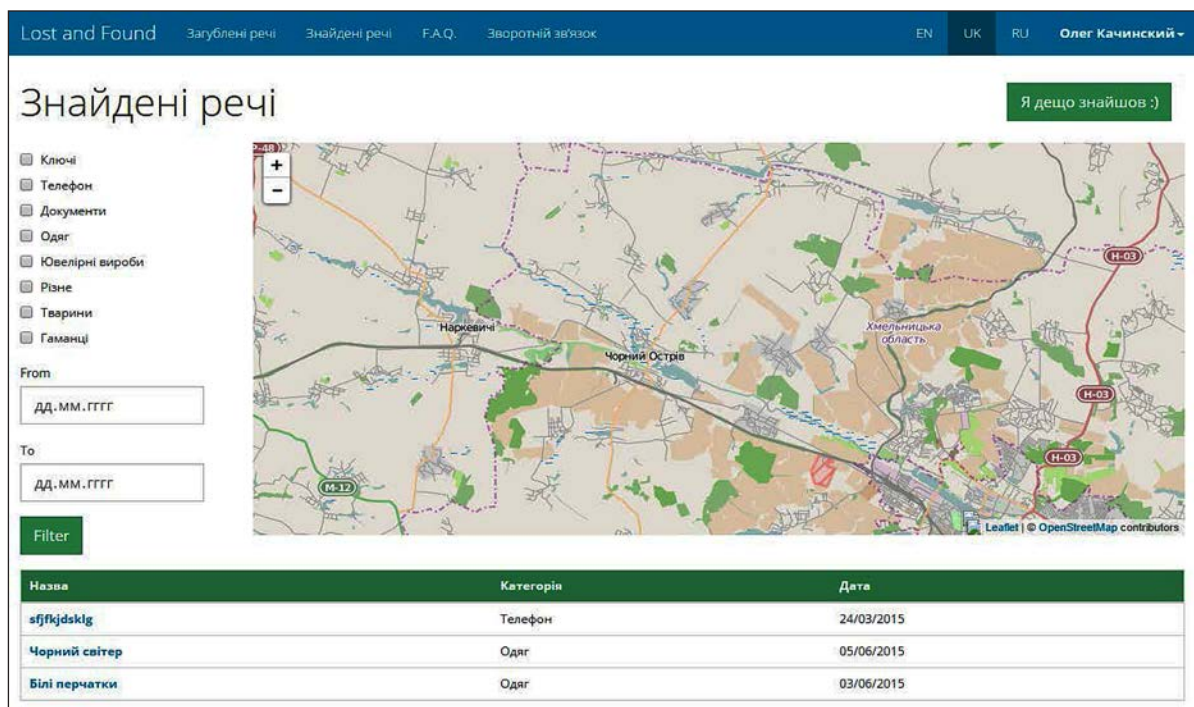


Рис. 6. Сторінка знайдених речей

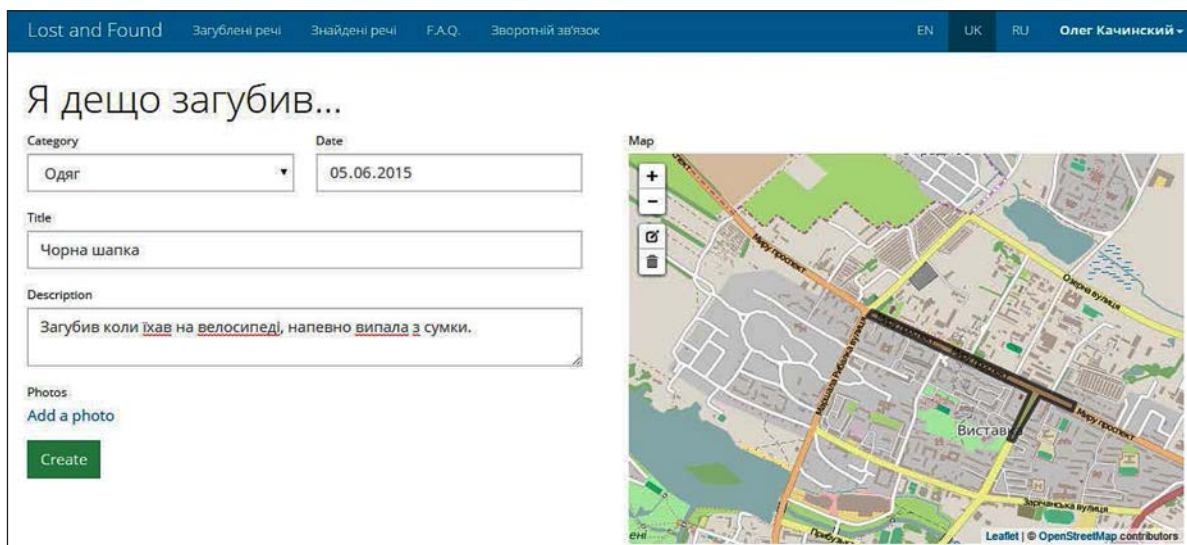


Рис. 7. Сторінка оголошення на загублену річ

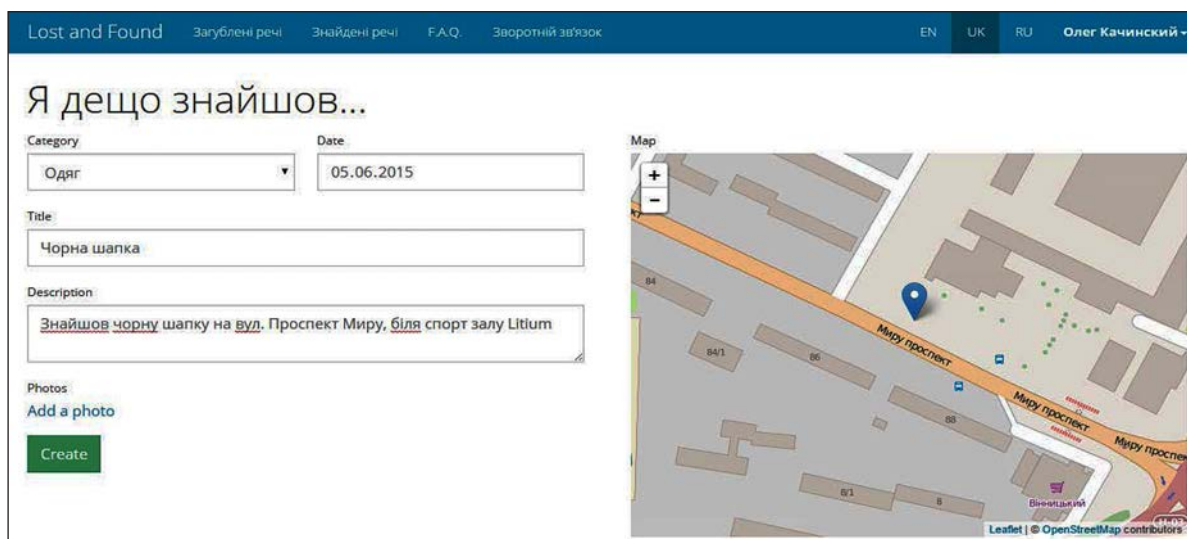


Рис. 7. Сторінка оголошення на знайдену річ

мають екземпляр класу менеджера, який керує всіма сутностями в системі. Кожний запит до бази даних відбувається з використанням вже доступних методів, які допомагають будувати запити.

Спочатку створюється екземпляр будівника запитів і далі в залежності від отриманих параметрів в методі виконується побудова запиту до бази даних. Будівник запитів має всі стандартні методи, які потрібні для побудови будь-якого запиту. Кожний елемент запиту є методом та більшість основних параметрів методи генерують самостійно завдяки цьому й досягається швидкодія побудови запитів необхідних для вибірки даних.

Перегляд предметів відбувається після того, як користувач перейде на відповідну сторінку знайдених або втрачених предметів, тобто виконується

дія, яка приймає запит користувача та дістає дані з бази. Така дія називається `foundItemsListAction` і виконується вона в контролері предметів.

Сторінку знайдених речей представлено на рис. 6.

Створення сторінки заявки відбувається за тим самим принципом, лише з використанням іншого методу.

На рис. 7 продемонстровано приклад додавання загубленої речі і вказування зони на карті.

Додавання знайденої речі із вказівкою точки на карті зображено на рис. 7

Після збереження оголошення в базі даних сервісу і валідації його адміністратором, користувачі отримують доступ до перегляду сторінки оголошення, яка зображена на рис. 8, з можливістю отримати контакти користувача, який додав ого-

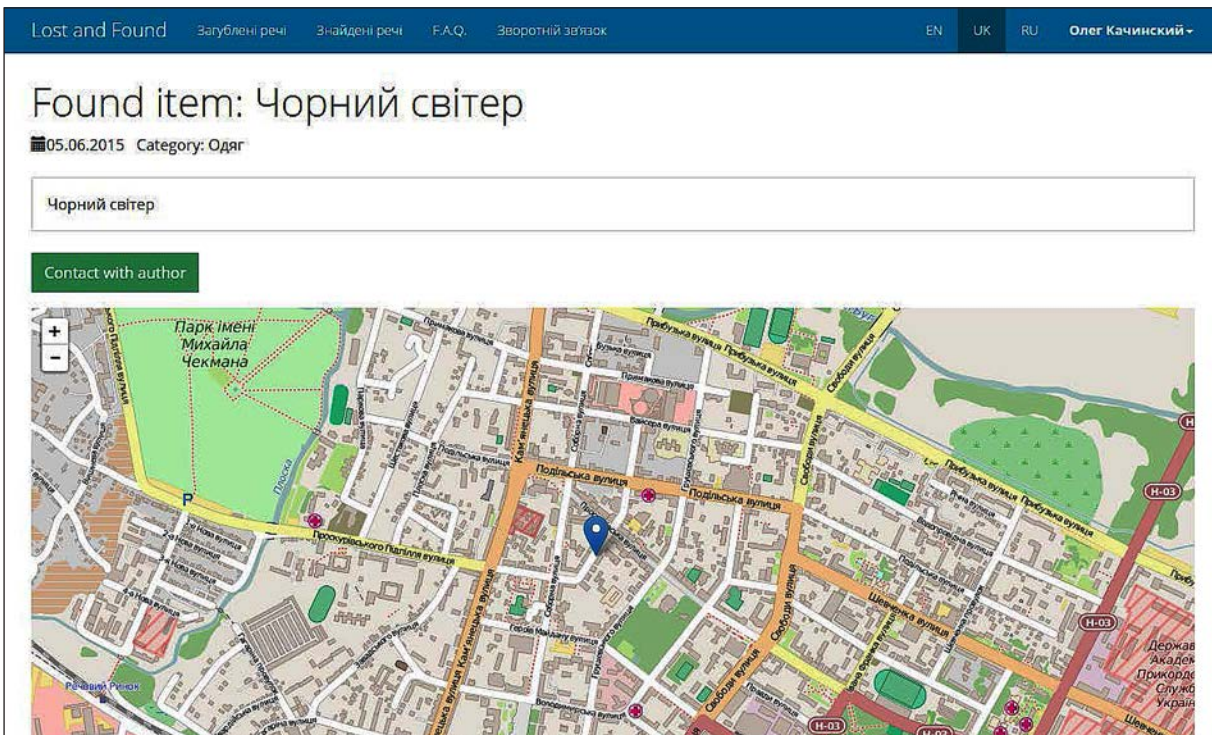


Рис. 8. Сторінка предмету

лошення (Facebook аккаунт) або з можливістю відправити йому повідомлення.

Після розробки було проведено тестування сервісу, яке допомогло виявити його недоліки та помилки. Після цього помилки було виправлено та веб-сервіс ще раз був наданий для тестування кільком реальним користувачам.

Висновки. У результаті проведеної роботи було створено потужний веб-сервіс для надання

послуг зі зберігання оголошень про втрачені та знайдені речі, зручним переглядом вже створених оголошень з можливістю фільтрації по категоріях, можливістю зв'язатися з автором оголошення. Основним завданням створеної системи є зменшення навантаження на органи муніципальної влади у роботі з предметами, які надходять до бюро знахідок, та створення веб-сервісу, який би був універсальний для всього світу.

Список літератури:

1. LostAndFound.com. URL: <https://www.lostandfound.com> (дата звернення: 01.10.2019).
2. UC Barkley Main Lost And Found. URL: <http://www.lib.berkeley.edu/libraries/doe-library/lost-and-found> (дата звернення: 01.10.2019).
3. Airport lost and found. URL: <https://www.airportslostandfound.com> (дата звернення: 01.10.2019).
4. ReturnMe. URL: <https://www.return.me> (дата звернення: 01.10.2019).
5. LuckFind. URL: <https://www.luckfind.me> (дата звернення: 01.10.2019).
6. VARTA1. URL: <https://www.911.lviv.ua> (дата звернення: 01.10.2019).
7. Документація та перелік можливостей скрипкової мови PHP. URL: <http://www.php.net> (дата звернення: 01.10.2019).
8. Документація та опис можливостей фреймворку Symfony. URL: <http://www.symfony.com/doc/current/index.html> (дата звернення: 01.10.2019).
9. Bootstrap. URL: <https://www.getbootstrap.com> (дата звернення: 01.10.2019).

Kotsun V.I. WEB SEARCHING SERVICE FOR FOUND AND LOST THINGS “LOST-AND-FOUND OFFICE”

The paper deals with the implementation of a web searching service for found and lost things. The web searching service was developed using PHP scripting language, Symfony framework, Javascript with jQuery library, Ajax technology, as well as HTML markup and CSS.

In today's society it is difficult to imagine a person without a standard set of personal belongings, such as keys, phone, wallet, pocket money, etc., but in a crazy life rhythm there is a high probability of losing an

important object. In most cases, small, important items such as keys, phone, passport, wallet and more are found, but it still happens that you have to use the help of various services and lost-and-found office to help you find important things. These services include finding offices, message boards, pawnshops, radio alerts, television, and more.

Existing search services for found and lost things and their main advantages and disadvantages are described and analyzed. The basic factors for the necessity of realization of the web search service for found and lost things are investigated. As one of the most widely used languages in web development, the PHP programming language was chosen to fulfill the developed system requirements.

The possibilities of creating a claim for a found or lost things, viewing existing claims, contacting the submitted applicants, authorizing through the Facebook social network, modifying applications, using a map to indicate the location of a found or lost things, obtaining statistics regarding created applications.

It is shown that the developed web search system's goal is to reduce the burden on municipal authorities in handling things coming to the lost-and-found offices and to develop the web search service as universal for the world requirements.

Key words: *web service, lost-and-found office, programming, framework, Javascript.*