

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-54-01>

УДК 004.031.4, 004.67, 37.09

Бубній Данило Юрійович, магістр

Дубук Василь Іванович, к.т.н., доцент

<https://orcid.org/0000-0002-6339-1032>

Національний університет «Львівська політехніка», м. Львів, Україна

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ПОШУКУ МЕНТОРА ДЛЯ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ В ІТ ГАЛУЗІ

Бубній Д.Ю., Дубук В.І. Розробка інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі. У статті обґрунтовано актуальність та наведено результати моделювання й розробки засобів інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі, які допоможуть здійснити об'єктивний та ефективний пошук ментора. Приведено опис, стисло характеристику та аналіз основних сервісних програмних засобів для пошуку ментора, а також визначено їх основні переваги і недоліки. У роботі розроблено та наведено концептуальну модель автоматизованої інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі, яка відображає спосіб взаємодії користувачів з елементами системи. Визначено та наведено інформаційні потоки в системі. Спроектовано основні компоненти системи та їх взаємозв'язки. Приведено діаграму випадків використання системи. Сформовано та відображено у вигляді діаграм основні функціональні можливості системи. Розглянуто всі сутності, які наявні в системі та зовнішні сервіси, з якими буде співпрацювати система. Розроблено алгоритм роботи автоматизованої інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі, а також діаграму прецедентів, контекстну діаграму та діаграму класів. Розроблено і наведено структуру бази даних системи пошуку ментора для підвищення кваліфікації в ІТ галузі. Розроблено та описано графічний людино-машинний інтерфейс системи, за допомогою якого можна організувати діалог користувача з системою та здійснити пошук ментора. Розроблену систему пошуку ментора для підвищення кваліфікації в ІТ галузі можна буде використовувати для ефективного пошуку ментора та цілісної співпраці, що сприятиме розвитку учнів для підвищення кваліфікації в ІТ галузі. Практичне застосування розробленої системи є актуальним і може сприяти подальшому розвитку освіти, кар'єрного зростання та самовдосконалення. Розроблена система забезпечує доступну платформу для знаходження відповідного ментора та сприяє підвищенню якості менторської підтримки, підвищенню рівня професійної освіти дорослих, що позитивно впливатиме на результати їх професійної діяльності.

Ключові слова: інформаційно-пошукова система, пошук ментора, графічний інтерфейс користувача, ІТ галузь, освіта дорослих.

Bubniy D., Dubuk V. Development of an information system for finding a mentor for advanced training in the IT industry. The article substantiates the relevance and gives the results of modeling and development of information system tools for finding a mentor for professional development in the IT industry, which will help to carry out an objective and effective search for a mentor. The description, concise characteristics and analysis of the main service software tools for finding a mentor are given, as well as their main advantages and disadvantages are determined. The work develops and presents a conceptual model of an automated information system for finding a mentor for professional development in the IT industry, which reflects the way users interact with system elements. Information flows in the system are defined and given. The main components of the system and their interrelationships are designed. A use-case diagram of the system is provided. The basic functionality of the system is formed and displayed in the form of diagrams. All entities available in the system and external services with which the system will cooperate are considered. The algorithm of the automated information system for finding a mentor for professional development in the IT industry, as well as a use-case diagram, a context diagram and a class diagram, have been developed. The database structure of the mentor search system for professional development in the IT industry has been developed and provided. A graphical human-machine interface of the system has been developed and described, with the help of which you can organize a user dialogue with the system and search for a mentor. The developed system of finding a mentor for advanced training in the IT industry can be used for effective mentor search and holistic cooperation, which will contribute to the development of students for advanced training in the IT industry. The practical application of the developed system is relevant and can contribute to the further development of education, career growth and self-improvement. The developed system provides an accessible platform for finding a suitable mentor and contributes to improving the quality of mentoring support, raising the level of professional education of adults, which will positively affect the results of their professional activities.

Keywords: information search system, mentor finding, graphical user interface, IT industry, adult education.

Постановка задачі. Розробка інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі є актуальним завданням в сучасному світі, де знання та підтримка стають ключовими факторами для досягнення успіху в різних сферах життя. Менторство, як механізм передачі досвіду та знань, грає важливу роль у розвитку професійної та особистісної кар'єри людини. Однак знаходження відповідного ментора та встановлення з ним зв'язку може бути завданням важким та займати багато часу. Інформаційна система пошуку ментора для підвищення кваліфікації в ІТ галузі може стати важливим інструментом для полегшення цього процесу.

У статті розглянуто важливість менторства та потребу в розробці інформаційної системи, яка допомагатиме людям знаходити підходящих менторів та спростувати комунікацію з ними.

Менторство є важливим елементом освіти дорослих, розвитку кар'єри та самовдосконалення, досвідчений ментор може надати значний внесок у розвиток індивіда.

Завданням відповідної інформаційної системи є створення платформи, де люди, які шукають ментора, зможуть знайти відповідних експертів у своїй галузі та встановити контакт з ними. Також система надаватиме можливість менторам пропонувати свої послуги та спілкуватися зі своїми учнями, сприяючи обміну знань та досвідом.

У цьому контексті розробка інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі стає важливим завданням, яке відповідає сучасним потребам і може сприяти подальшому розвитку освіти та кар'єрного зростання. У наш час, коли швидкість розвитку технологій робить знання ще більш доступними, інформаційна система цього типу може стати ключовим інструментом для тих, хто прагне досягнути успіху у своїй професійній діяльності.

Предметна область – пошук ментора для підвищення кваліфікації в ІТ галузі.

Актуальність дослідження визначається рядом факторів, які впливають на сучасне суспільство та його потреби. Швидкий розвиток технологій змінив спосіб, яким ми отримуємо доступ до знань і шукаємо інформацію. Постійне навчання стало нормою, тому менторство може покращити якість цього процесу та зробити його більш доступним. Менторство дозволяє учням отримувати індивідуальну увагу та підтримку в процесі навчання. Розвиток кар'єри вимагає підтримки та керівництва, які можуть надати ментори. Глобальна комунікація робить можливим співпрацю з менторами з різних країн, розширюючи можливості навчання та розвитку. Усі ці фактори підкреслюють актуальність створення інформаційної системи для менторства, яка сприятиме як особистісному, так і професійному зростанню людей.

Метою дослідження є розробка інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі. Також потрібно зазначити, що таку систему можна використовувати як автономно, так й інтегрувати із існуючими сервісами всередині компанії.

У цій науковій роботі *об'єктом дослідження* є процес пошуку ментора для підвищення кваліфікації в ІТ галузі.

Предметом дослідження даної теми є моделювання та розробка інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі.

Задачі роботи системи: зберігання інформації про ментора; зберігання інформації про користувача, що шукає ментора; пошук ментора за допомогою фільтрів (країна, досвід, набір технологій та інше); створення календаря для призначення зустрічей з ментором чи учнем; відображення навиків учня для діагностики та відслідковування його прогресу; можливість надіслати запит на менторство, який надалі можна прийняти чи відхилити; можливість знайти ментора використовуючи функціональні можливості експертної системи; можливість відслідковувати статус запиту на менторство; створення особистого профіля з можливістю його редагувати.

Науковий результат – розробка інформаційної системи, яка сприятиме знаходженню менторів та покращить комунікацію між менторами і учнями, розвитку та удосконаленню їх співпраці. При цьому удосконалено метод організації навчання та виховання й консультування з урахуванням індивідуальних особливостей учасників навчального процесу з використанням сучасних інформаційної системи та інформаційних технологій.

Практична цінність досліджень — розроблено інформаційну систему, яка сприятиме знаходженню відповідних менторів та полегшить взаємодію між менторами і учнями, сприяючи особистісному та професійному зростанню останніх.

Проблема дослідження – моделювання та розробка інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі. Для вирішення проблеми пошуку та надання менторської підтримки в сучасному світі вже використовуються різні засоби і методи, хоча інформаційні системи для цього ще не так поширені, як можна було б очікувати.

Аналіз останніх досліджень і публікацій. Наведемо вибрані приклади засобів та мето-дів, які вже працюють в обраній предметній області:

- *Онлайн-платформи.* Деякі платформи, такі як LinkedIn, Meetup, або MentorCity, дозволяють користувачам знаходити менторів і спілкуватися з ними в онлайн-режимі.

- *Спеціалізовані форуми та соціальні мережі.* Вони дозволяють користувачам знайти і організувати спілкування з потенційними менторами або учнями, якщо вони шукають допомогу або надають її в певних галузях.

- *Менторські програми в університетах та організаціях.* Багато навчальних закладів та підприємств запускають власні менторські програми, які допомагають знаходити менторів та учнів, використовуючи для цього корпоративну інформаційну систему.

- *Пошукові системи та веб-сайти.* Використання пошукових систем для знаходження ресурсів та засобів, які допомагають в розвитку, або для пошуку експертів у певних галузях.

- *Консультації і тренінги.* Консультації та тренінги, надані в реальному режимі, допомагають учням знайти ментора та отримати реальну підтримку.

- *Саморозвиток і навчання в мережі.* Можливість навчання та самостійного розвитку через відкриті онлайн-курси, підручники та відеоуроки.

- *Професійні асоціації та групи спільності.* Ці організації нерідко організують зустр-рчі та заходи, на яких можна знайти менторів або стати частиною менторської програми.

Однак, здійснивши аналіз вище вказаних засобів та методів, які працюють в обраній предметній області, для багатьох із них було виявлено наявність проблем забезпечення належного рівня захисту інформації, обумовленого номативно-правовим полем [1].

Розглянемо один існуючий аналог системи від компанії Foxminded — це платформа, що надає навчання в галузі IT та програмування [2]. Їх основна унікальність полягає у використанні методу «mentoring by practice», який спрямований на практичне застосування знань у реальних проектах.

Наведемо деякі основні переваги та недоліки платформи Foxminded для менторства.

Переваги:

1. *Практична спрямованість.* Основна особливість Foxminded — це навчання через роботу над реальними проектами. Це дозволяє учням отримати не лише теоретичні знання, а й практичний досвід, що важливо у галузі IT.

2. *Менторство.* Кожен учень отримує особистого наставника, який допомагає у вирішенні завдань та надає консультації під час проходження курсу. Це сприяє ефективному засвоєнню матеріалу та розвитку навичок.

3. *Широкий вибір курсів.* Foxminded пропонує різноманітні програми та курси в галузі програмування, робототехніки, інженерії програмного забезпечення тощо, що дає можливість знайти відповідний курс для своїх потреб.

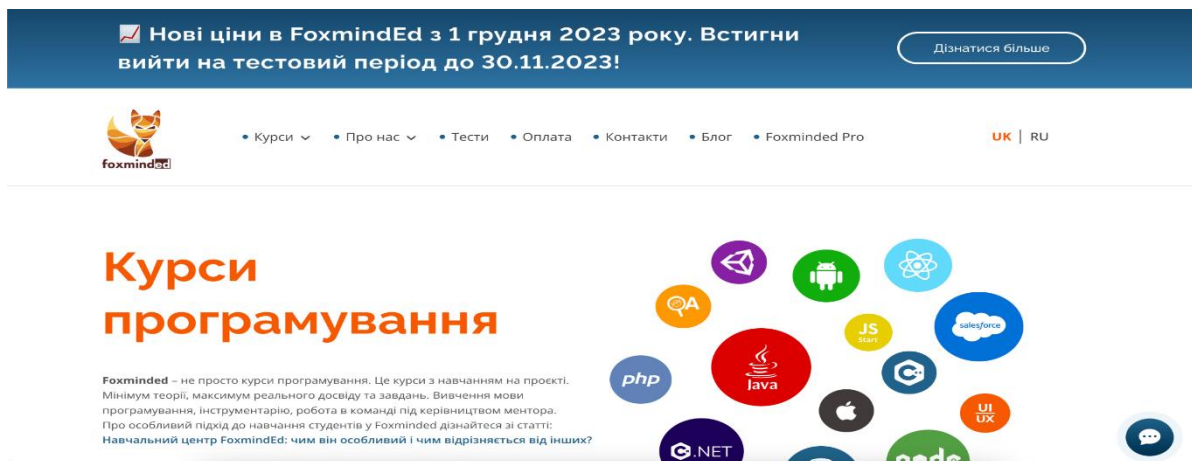


Рис. 1 – Головна сторінка компанії Foxminded

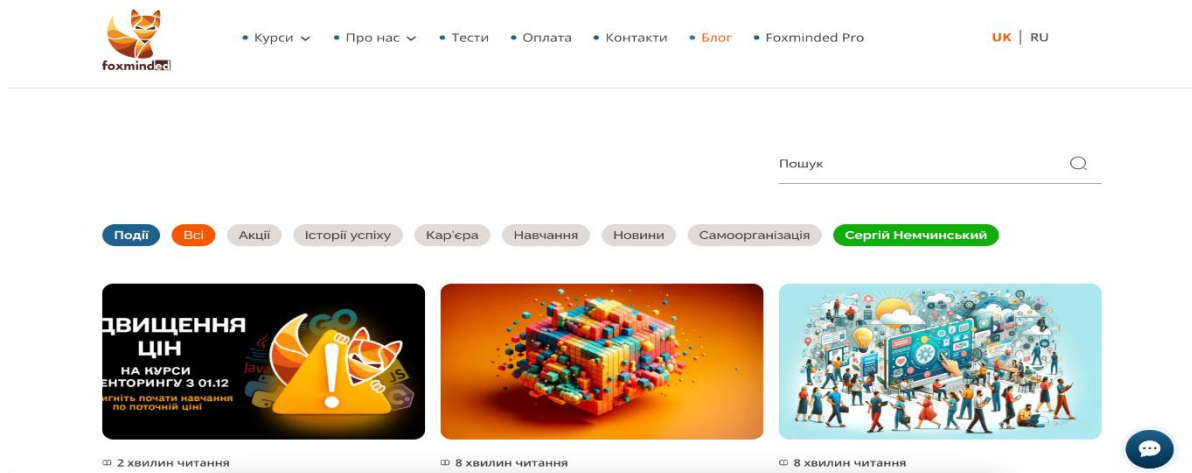


Рис. 2 – Сторінка «Блог» компанії Foxminded

4. *Стажування та можливості працевлаштування.* Після завершення курсу учні мають можливість отримати стажування чи працевлаштуватися у компаніях-партнерах Foxminded.

Недоліки:

1. *Вартість.* В порівнянні з деякими іншими платформами навчання, вартість курсів Foxminded може бути високою, що робить їх недоступними для деяких людей.

2. *Обмежена географія.* Наразі Foxminded може мати обмежену географію дії, тобто може бути недоступним у деяких регіонах.

3. *Нестабільність програми навчання.* Хоча концепція навчання через практику дуже цінна, іноді програми можуть бути нестабільними або могли потребувати певного уточнення у матеріалах.

В цілому, Foxminded пропонує цікавий підхід до навчання, зосереджений на практичних вміннях та особистому менторстві, але варто врахувати індивідуальні потреби та можливості перед вибором цієї платформи.

Хоча ці існуючі засоби і методи можуть бути корисними, розробка інформаційної системи для менторства може спростити та поліпшити процес пошуку та взаємодії між менторами та учнями, забезпечуючи більшу доступність та ефективність цієї важливої складової освіти та розвитку дорослих.

Основна частина. Першим кроком у моделювання та розробку інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі є побудова концептуальної моделі. Відповідна модель дає змогу передбачити роботу всіх користувачів системи та їх взаємодію з елементами системи. Концептуальна модель інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі зображена на рис.3.

Основні потоки включають наступне: авторизацію користувача; підбір та пошук ментора; створення та маніпуляцію з навичками; відображення навиків учня та менторів в JSON форматі, з використанням JavaScript бібліотек [3]; відображення та створення подій в календарі.

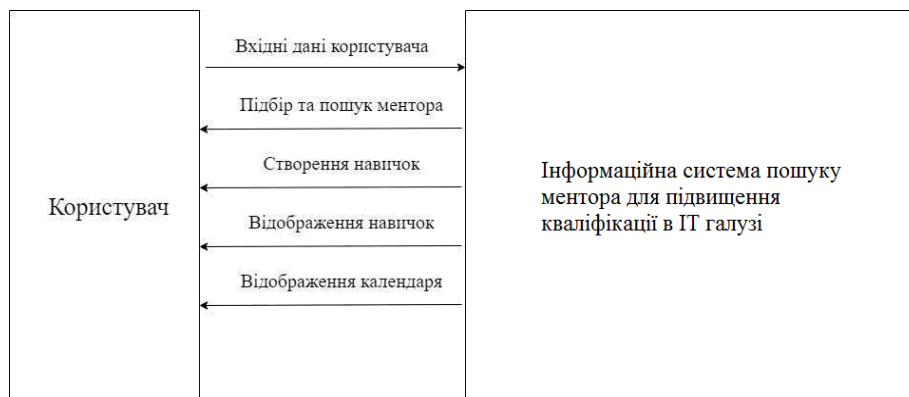


Рис. 3 – Концептуальна модель інформаційної системи

Огляд моделі експертної системи та блок-схеми алгоритмів. Експертна система — реалізована засобами програмного забезпечення, яке моделює та робить висновки на основі експертного знання в конкретній галузі. Алгоритми роботи експертної системи включають в себе різні кроки і процедури, які допомагають системі аналізувати проблему та надавати рекомендації або рішення. Наведемо деякі загальні алгоритми, які використовуються в експертних системах.

Захоплення знань. Цей крок полягає в тому, щоб зібрати експертне знання від людей, які спеціалізуються в конкретній галузі. Інформація може бути представлена у вигляді правил, фактів, евристик та інших форм.

Подання знань. Експертні системи повинні мати внутрішню структуру для представлення зібраного знання. Це може включати в себе використання правил, семантичних мереж, продукційних систем та інших методів для організації та представлення знань.

Інференція. Цей алгоритм використовується для виконання логічних висновків на основі доступного знання. Експертна система використовує цей алгоритм для вироблення рішень або надання рекомендацій на основі питань та фактів, які користувач вводить.

Виведення. Експертна система може надавати користувачам пояснення своїх рішень або рекомендацій. Це може бути корисним для розуміння того, як експертна система прийшла до певного висновку. Вказано лише загальні кроки та алгоритми, які використовуються в експертних системах.

Захист інформації. Система враховує вимоги до захисту персональних даних користувачів згідно норм, визначених [1], та не зберігає їх для власного використання.

У випадку моделювання та розробки засобів інформаційної системи пошуку ментора для підвищення кваліфікації в IT галузі, моделлю власне виступає експертна система, яка працює наступним чином: користувач, коли обирає ментора, має можливість зробити це використовуючи засоби штучного інтелекту, тобто експертну систему, яка побудована у вигляді запитань, на які користувач поступово надає відповіді «так» чи «ні».

Наприклад: «Чи цікавить Вас розробка користувацького інтерфейсу?». Відповідь: «так». В цьому випадку система, базуючись на відповіді користувача, сформує наступні уточнюючі запитання в потрібному напрямку, відкинувши інші протилежні за змістом. Також питання враховують інформаційну психологію спілкування [4].

Таким чином, коли користувач дасть відповідь на останнє запитання, в результаті буде підібраний ментор, якого експертна система підбрала базуючись на відповідях користувача. Кожна експертна система ґрунтується на базі знань, що складається з правил, фактів та інших елементів.

У розробленій системі база знань, заповнювалась за допомогою експертів, які надавали свої знання, визначали правила та встановлювали зв'язки між ними. Такий підхід дозволяє системі приймати виражені рішення на основі зібраних експертних знань.

Було розроблено блок-схеми алгоритму роботи експертної системи та пошуку ментора, які зображені на рис.4 та рис.5. Всі рисунки були побудовані у програмі Draw.io [5].

Користувач при переході на сторінку пошуку ментора матиме змогу виконати пошук ментора самостійно за допомогою набору фільтрів або скористатись функціональністю експертної системи, тобто система підбере ментора найбільш ефективно.

На рис 5. відображено другий важливий алгоритм, який дозволяє знайти ментора. Спочатку користувачеві пропонується прийняти рішення - чи довірити системі знайти ментора автоматично, чи знайти його самостійно за допомогою набору фільтрів.

Якщо користувач вибирає підібрати ментора автоматично, тоді система пропонує дати відповідь на ряд запитань, за допомогою яких буде підбрано ментора найбільш ефективно.

Якщо користувач вибирає знайти ментора самостійно, тоді система відображає спеціальні поля, де можна налаштувати певні фільтри, щоб знайти потрібного ментора.

Діаграма випадків використання - це особлива форма системних програмних вимог до нової слабозрозуміної програми. Ключова концепція моделювання випадків використання полягає в тому, що це допомагає розробити систему з точки зору кінцевого користувача [6]. Це ефективна техніка передачі поведінки системи з точки зору користувача шляхом визначення всієї видимої зовні системи поведінки [7].

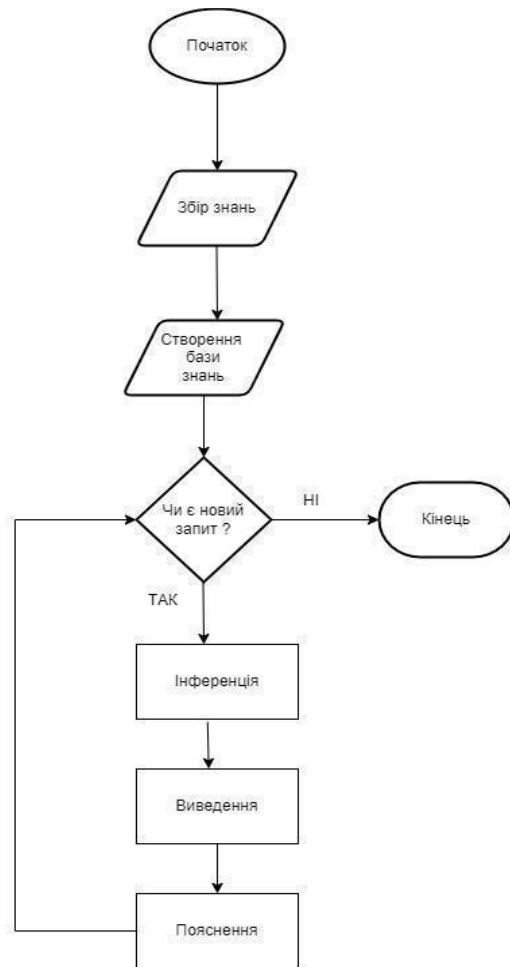


Рис. 4 – Блок схема алгоритму роботи системи



Рис. 5 – Блок схема алгоритму пошуку ментора

На даній діаграмі відображено три види користувачів: користувач, ментор та адміністратор. Для розподілення ролей та надання доступу до ресурсів в системі використовувались бібліотека Spring Security [8].

Діаграма послідовностей – це діаграма, яка показує взаємодію об'єктів за часом. Тобто відображаються дії користувача протягом часу використання системи.

Використання діаграм послідовностей є ключовим для уявлення та розуміння взаємодії об'єктів в системі протягом часу. Ця діаграма відображає послідовність подій та повідомлень між об'єктами системи під час їх взаємодії. Наприклад, для інформаційної системи пошуку ментора в ІТ галузі на діаграмі послідовностей можуть бути описані такі важливі етапи, як створення облікового запису користувача, процес пошуку ментора та вихід з системи. На цій діаграмі послідовностей буде відображено, як користувач інтерактивно взаємодіє з системою на кожному кроці, починаючи зі створення облікового запису користувача, через введення критеріїв пошуку ментора, до виходу з системи після завершення сеансу. Це дозволить краще зрозуміти та оптимізувати процеси взаємодії користувача з інформаційною системою та забезпечить ефективність її використання.

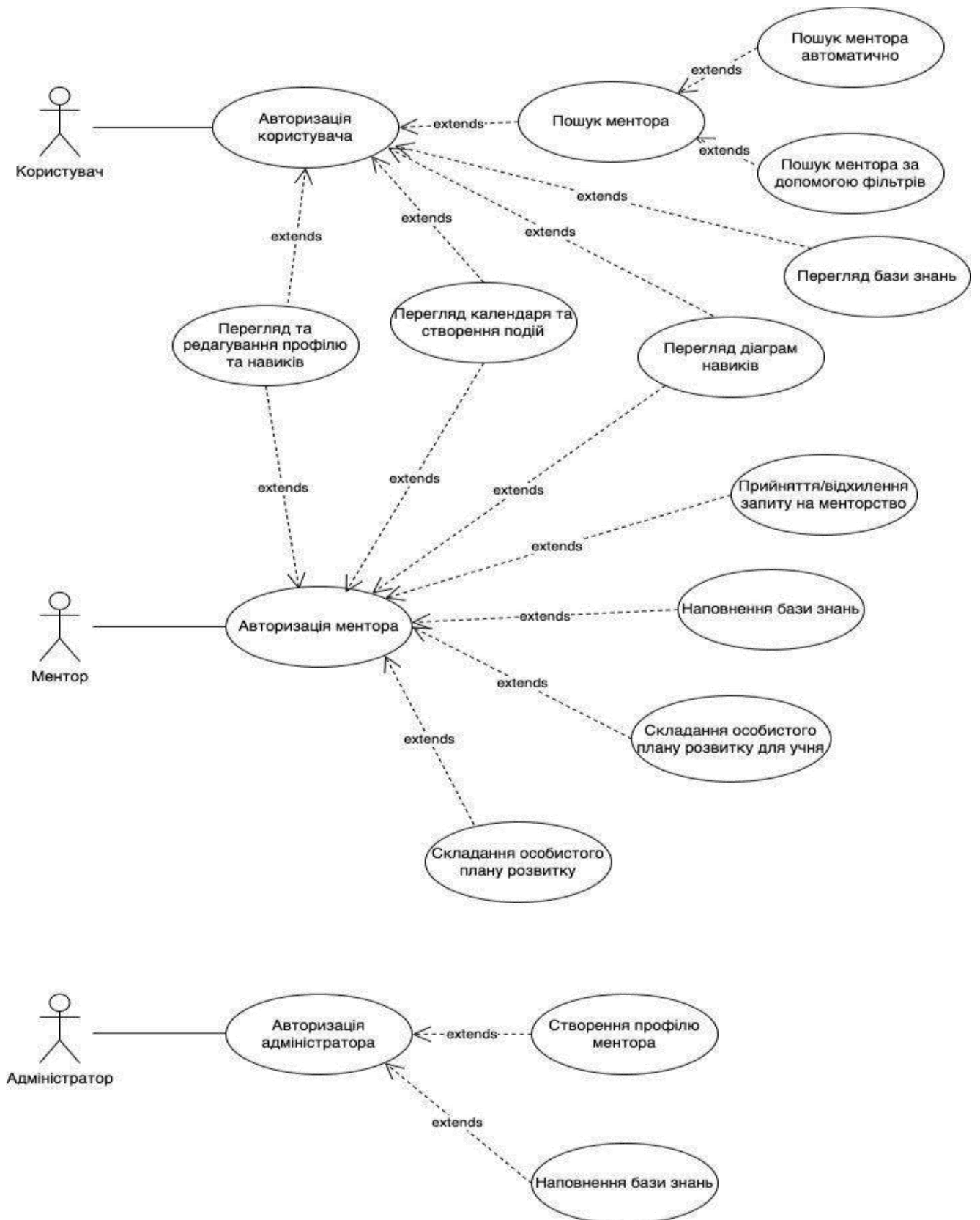


Рис. 6 – Діаграма випадків використання системи

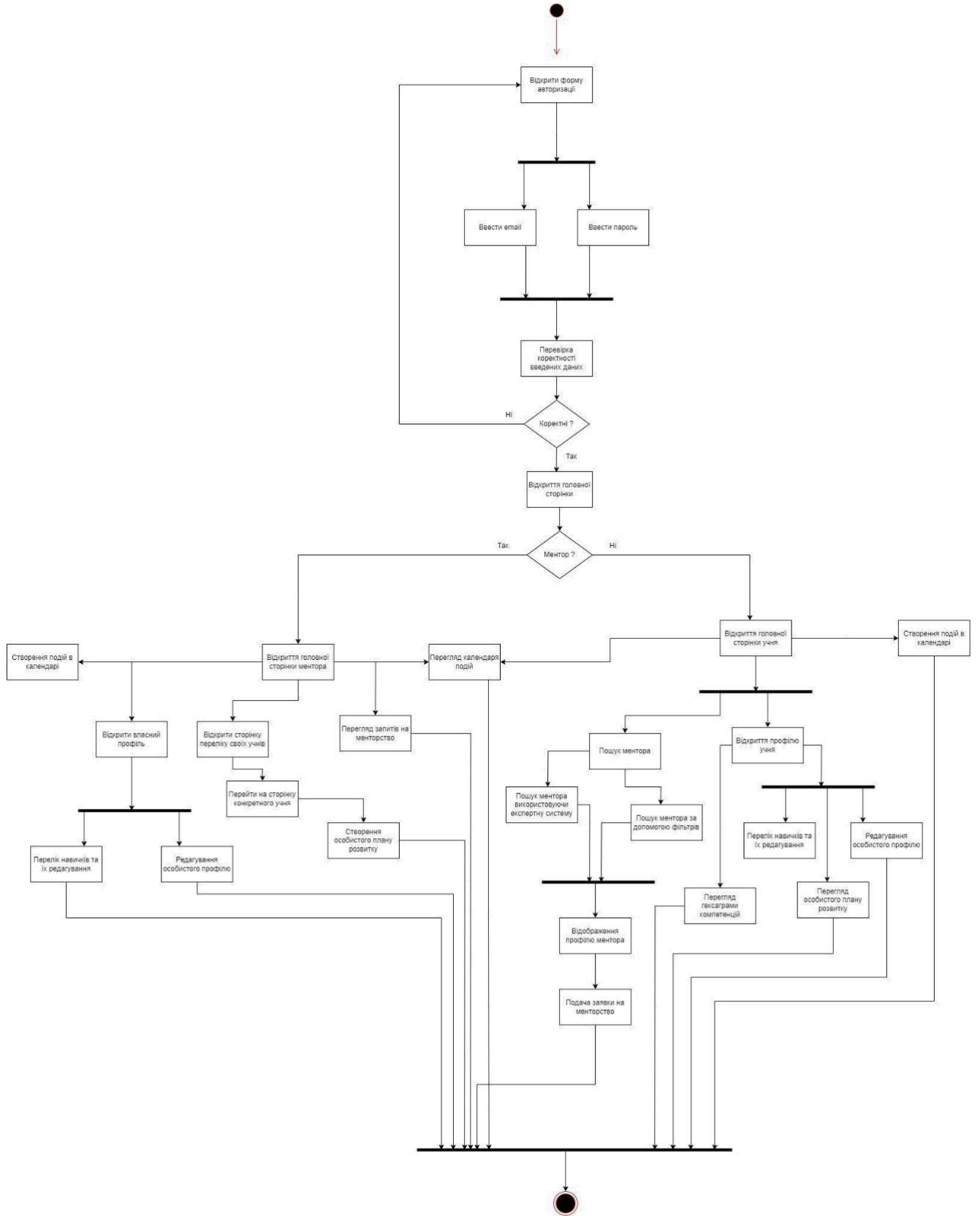


Рис. 7 – Діаграма діяльностей для системи

На рис.8 представлено послідовності для створення запису користувача, пошук ментора та вихід з системи. За основу використовувалась архітектура MVC [9], [10].

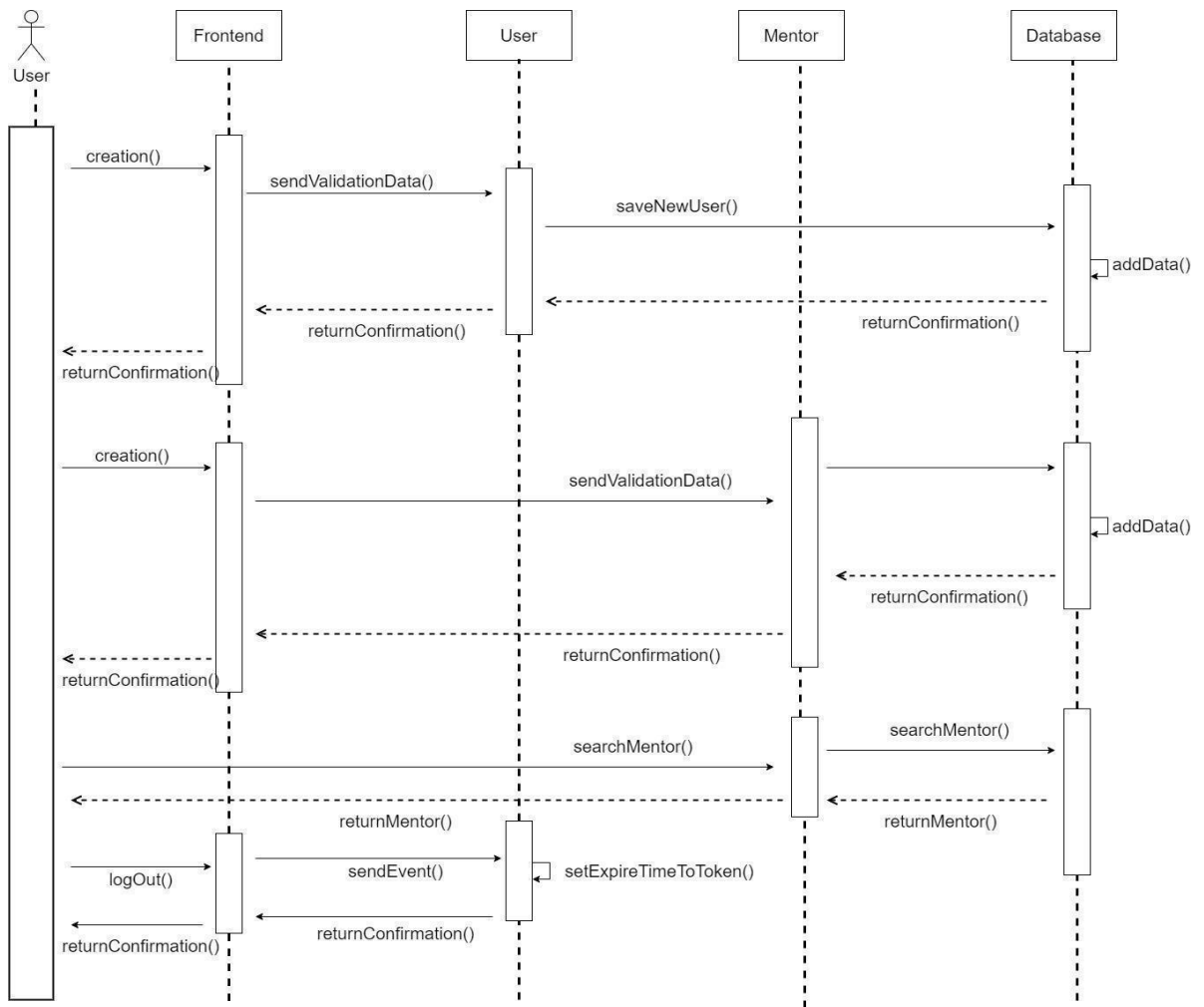


Рис. 8 – Діаграма послідовностей

Наступною розробленою діаграмою системи є діаграма класів, представлена на рис. 9.

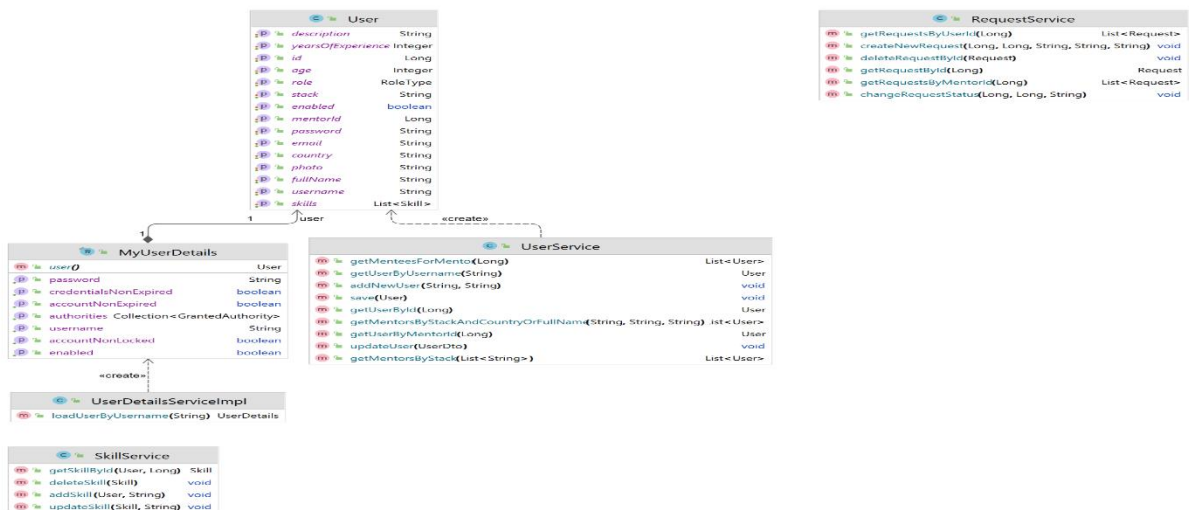


Рис. 9 – Діаграма класів системи

Структура даних, відображена схемою бази даних, зображена на рис.10, яка маніпулюється за допомогою серверної частини, що написана на мові програмування Java [11] та технології Spring Boot [12].

До головних сутностей можна віднести: «users», «roles» та «skill»А». Головні сутності мають

також допоміжні таблиці з додатковою інформацією про них. Дані з таблиці «skills» візуалізуються в графіки для кращого розуміння прогресу учня [13].

Інформація про користувача несе в собі персональні дані про менторів та учнів. Таблиця також має поле поштової скриньки задля надсилання підтвердження щодо реєстрації, а також отримання інформації щодо процесу менторства. Кожен користувач відповідно прив'язаний до ментора. Основний функціонал виконують таблиці: «users», «skill», «roles» та «goals». Для зв'язування бази даних, а саме PostgreSQL [14] з серверною частиною використовувалась технологія Hibernate [15], що згодом була замінена на Spring Data JPA [16], так як вона забезпечує більш високий рівень автоматизації розробки.

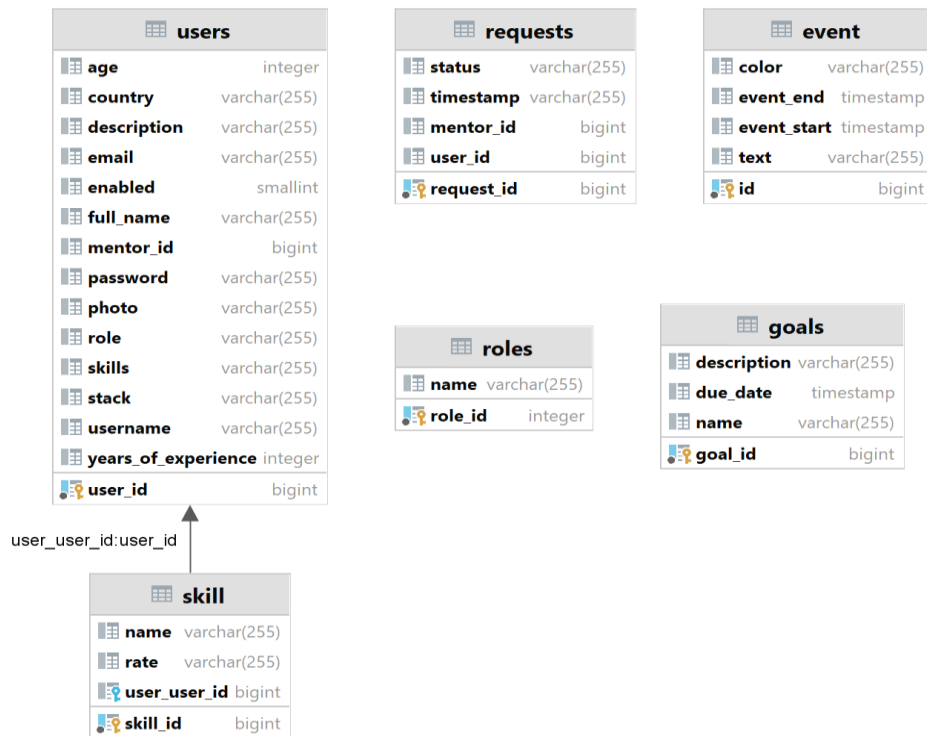


Рис. 10 – Схема бази даних системи

Експертна підсистема в даній системі працює у вигляді блоку опитування, який підбирає менторів для учнів найбільш ефективно. Дане опитування містить 10 різних запитань. Відповіді зберігає програма під час виконання і згодом їх використовує для отримання результату. Також деякі дані побудовані у вигляді графів, та для ефективного шляху проходження графу було задіяно алгоритм Йозефа Бернарда Крускала [17].

Далі розглянемо сторінку «Знайти ментора». Тут користувачу пропонується, ввести ім'я та прізвище ментора або виконати пошук застосовуючи фільтри, такі як Технологія та Країна. Також користувач може скористатись функціональними можливостями експертної системи, натиснувши на кнопку «Не знаєш як вибрати ментора? Тобі сюди...». Інтерфейс для взаємодії з користувачем був розроблений використовуючи технології CSS [18] та Angular [19], Bootstrap [20].

Якщо користувач вирішує знайти ментора за допомогою експертної системи, йому слід відповісти на ряд запитань, що допоможуть у цьому. Загальний вигляд сторінки представлений на рис. 11.

Також було додано детальний опис ресурсів з якими можуть взаємодіяти користувачі, використовуючи технологію Swagger [21].

Під час розробки системи було враховані нові тенденції веб додатків [7], [22] на ринку програмного забезпечення [23].

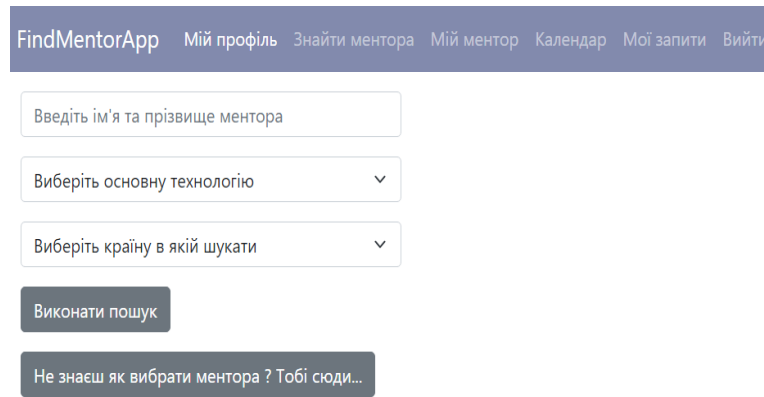


Рис. 11 – Видгляд сторінки «Знайти ментора»

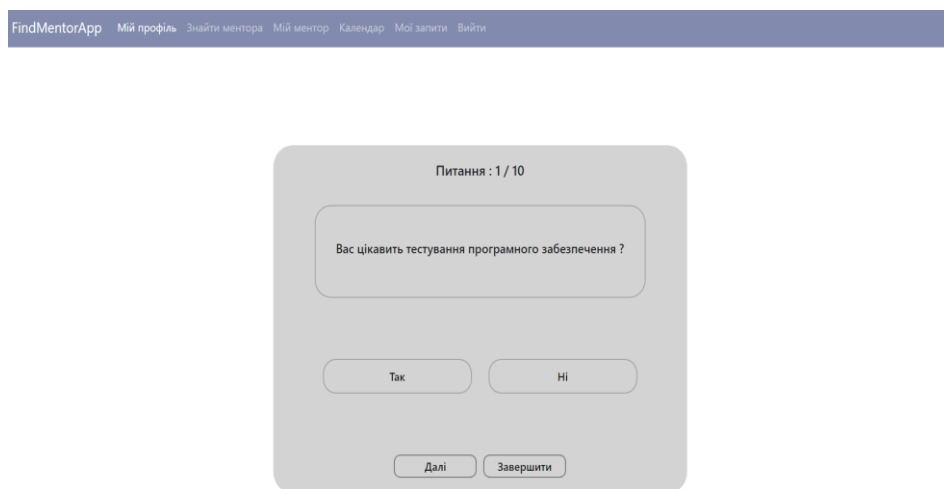


Рис. 12 – Видгляд сторінки «Форма діалогу з користувачем»

Висновки. У результаті виконаної роботи визначено основні ознаки та функції інформаційної системи пошуку ментора для підвищення кваліфікації в ІТ галузі. Було знайдено подібні рішення, розкрито їх переваги та недоліки. Як результат, було розроблено інформаційну систему пошуку ментора для підвищення кваліфікації в ІТ галузі.

У процесі роботи було визначено характеристики об'єкта дослідження та сформульовано подальші завдання, які необхідно виконати. Було проведено огляд літератури, описано інструменти та методи, які було використано для вирішення заданої проблеми. Також було розглянуто подібні системи та описано вибрані методи розробки програмного забезпечення. Проаналізовано та вибрано методи і розроблено алгоритми, які ефективно застосовні для вирішення проблем.

Розроблені та продемонстровані моделі роботи системи, включаючи діаграми UML, DFD, IDEF, а також діаграму класів і схему бази даних.

Як результат, розроблено інформаційну систему, яка сприятиме знаходженню менторів та покращить комунікацію між менторами і учнями, розвитку та удосконаленню їх співпраці. При цьому удосконалено метод організації навчання та виховання й консультування з урахуванням індивідуальних особливостей учасників навчального процесу з використанням сучасних інформаційної системи та інформаційних технологій.

Розроблено інформаційну систему, яка сприятиме знаходженню відповідних менторів та полегшить взаємодію між менторами і учнями, сприяючи особистісному та професійному зростанню останніх.

Розроблена інформаційна система пошуку ментора для підвищення кваліфікації в ІТ галузі повністю готова до роботи в організаціях і відкрита для подальшого функціонального розширення та підтримки.

У майбутньому, передбачено можливість удосконалення розробленої системи в напрямку

розширення виконуваних функцій та її адаптації до роботи з системами на основі мобільних платформ.

Список бібліографічного опису

1. Про захист персональних даних: Закон України від 01.06.2010 р. № 2297-VI. / Верховна Рада України. URL: <https://zakon.rada.gov.ua/laws/show/2297-17> (дата звернення: 28.02.2024).
2. Курси програмування. Foxminded. URL: <https://foxminded.ua/> (дата звернення: 28.02.2024).
3. Флэнаган Д. JavaScript. Полное руководство. СПб.: Диалектика, 2021. 720 с.
4. Зінченко В. Соціоніка - інформаційна психологія спілкування. Освіта.ua. URL: <https://osvita.ua/school/method/teacher/3748> (дата звернення: 28.02.2024).
5. Security-first diagramming for teams. Bring your storage to our online tool, or save locally with the desktop app. Draw.io. JGraph Ltd. URL: <https://www.drawio.com/> (date of access: 01.03.2024).
6. Fowler M., Kendall S. UML Distilled Second Edition A Brief Guide to the Standard Object Modeling Language. Boston: Addison Wesley, 1999. 224 p.
7. 15 провідних тенденцій веб розробки у 2020 році. Web4U. URL: <https://web4u.in.ua/blog/15-prov-dnih-tendenciy-veb-rozrobki-u-2020-roc-31> (дата звернення: 01.03.2024).
8. Spring Security 6.2.2. Spring by VMware Tanzu. URL: <https://spring.io/projects/spring-security> (date of access: 01.03.2024).
9. Codecademy Team. MVC: Model, View, Controller. Codecademy. URL: <https://www.codecademy.com/article/mvc> (date of access: 01.03.2024).
10. Spring MVC Tutorial. JavaTpoint. Sonoo Jaiswal. URL: <https://www.javatpoint.com/spring-mvc-tutorial> (date of access: 01.03.2024).
11. Блох Д. Java: эффективное программирование. СПб.: Диалектика, 2019. 464 с.
12. Spring Boot - Introduction. Tutorials Point India Private Limited. URL: https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm (date of access: 28.02.2024).
13. Як і для чого використовувати візуалізацію даних? Ейдос: центр політичних студій та аналітики. URL: <http://eidodos.org.ua/novyny/yak-i-dlya-choho-vykorystovuvaty-vizualizatsiyu-danyh/> (дата звернення: 28.02.2024).
14. PostgreSQL: The World's Most Advanced Open Source Relational Database. The PostgreSQL Global Development Group. URL: <https://www.postgresql.org/> (date of access: 01.03.2024).
15. Hibernate Tutorial. JavaTpoint. Sonoo Jaiswal. URL: <https://www.javatpoint.com/hibernate-tutorial> (date of access: 01.03.2024).
16. Spring Data JPA 3.2.3. Spring by VMware Tanzu. URL: <https://spring.io/projects/spring-data-jpa> (date of access: 01.03.2024).
17. Алгоритм Крускала. Algoua. URL: https://algoua.com/algorithms/graphs/mst_kruskal/ (дата звернення: 01.03.2024).
18. CSS: Cascading Style Sheets. Mozilla. URL: <https://developer.mozilla.org/en-US/docs/Web/CSS> (date of access: 28.02.2024).
19. Introduction to Angular concepts. Angular. Google. URL: <https://angular.io/guide/architecture> (date of access: 28.02.2024).
20. Build fast, responsive sites with Bootstrap. Bootstrap. Bootstrap Team. URL: <https://getbootstrap.com/> (date of access: 01.03.2024).
21. API Development for Everyone. Swagger. SmartBear Software. URL: <https://swagger.io/> (date of access: 01.03.2024).
22. Snyder K. Website Development: A Beginner's Guide. Forbes Advisor. Forbes Media LLC. URL: <https://www.forbes.com/advisor/business/website-development/> (date of access: 01.03.2024).
23. Ринок розробки програмного забезпечення. PRO Платформа ефективного регулювання. Міністерство економіки України. URL: <https://regulation.gov.ua/dialogue/it-i-telekom/14-rinok-rozrobki-programnogo-zabezpesenna> (дата звернення: 01.03.2024).

References

1. Zakon Ukrainy Pro zakhyst personalnykh danykh: pryiniaty 1 cherv. 2010 roku, № 2297-VI / [Law of Ukraine on Defence of Personal Data from June 1, 2010, № 2297-VI]. Verkhovna Rada Ukrainy - Supreme Council of Ukraine. Retrieved from: <https://zakon.rada.gov.ua/laws/show/2297-17> [in Ukrainian].
2. Kursy programuvannya. [Courses of Programming]. Foxminded. (2024). URL: <https://foxminded.ua/>
3. Flanagan, D. (2021). JavaScript: The Complete Guide. Saint-Petersburg: Dialektika. [in Russian].
4. Zinchenko, V. (2009). Sotsionika – informatsijna psyhologija spilkuvannya. [Socionics – information psychology of intercommunication]. Osvita.ua. Retrieved from: <https://osvita.ua/school/method/teacher/3748> [in Ukrainian].
5. Security-first diagramming for teams. Bring your storage to our online tool, or save locally with the desktop app. Draw.io. JGraph Ltd. (2024). Retrieved from: <https://www.drawio.com/>
6. Fowler, M. & Kendall, S. (1999). UML Distilled Second Edition A Brief Guide to the Standard Object Modeling Language. Boston: Addison Wesley.
7. 15 providnykh tendencij web rozrobky u 2020 rotsi [15 leading trends of Web development in 2020]. Web4U. (2020). URL: <https://web4u.in.ua/blog/15-prov-dnih-tendenciy-veb-rozrobki-u-2020-roc-31>
8. Spring Security 6.2.2. Spring by VMware Tanzu. (2024). Retrieved from: <https://spring.io/projects/spring-security>
9. Codecademy Team. MVC: Model, View, Controller. Codecademy. (2024). Retrieved from: <https://www.codecademy.com/article/mvc>
10. Spring MVC Tutorial. JavaTpoint. Sonoo Jaiswal. (2024). Retrieved from: <https://www.javatpoint.com/spring-mvc>

tutorial

11. Bloch, J. (2019). Java: Efficient Programming. Saint-Petersburg: Dialektika. [in Russian].
12. Spring Boot - Introduction. Tutorials Point India Private Limited. (2024). Retrieved from: https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm
13. Як і для чого використовувати візуалізацію даних? [How and what to use data visualization?]. Eidos: Tsentr Politychnykh Studij ta Analityky – Eidos: Centre of Political Studies and Analytics. (2014). Retrieved from: <http://eidos.org.ua/novyny/yak-i-dlya-choho-vykorystovuvaty-vizualizatsiyu-danyh/> [in Ukrainian].
14. PostgreSQL: The World's Most Advanced Open Source Relational Database. The PostgreSQL Global Development Group. (2024). Retrieved from: <https://www.postgresql.org/>
15. Hibernate Tutorial. JavaTpoint. Sonoo Jaiswal. (2024). Retrieved from: <https://www.javatpoint.com/hibernate-tutorial>
16. Spring Data JPA 3.2.3. Spring by VMware Tanzu. (2024). Retrieved from: <https://spring.io/projects/spring-data-jpa>
17. Алгоритм Крускала [Kruskal algorithm]. Algoua. (2023). Retrieved from: https://algoua.com/algorithms/graphs/mst_kruskal/ [in Ukrainian].
18. CSS: Cascading Style Sheets. Mozilla. (2024). Retrieved from: <https://developer.mozilla.org/en-US/docs/Web/CSS>
19. Introduction to Angular concepts. Angular. Google. (2023). Retrieved from: <https://angular.io/guide/architecture>
20. Build fast, responsive sites with Bootstrap. Bootstrap. Bootstrap Team. (2024). Retrieved from: <https://getbootstrap.com/>
21. API Development for Everyone. Swagger. SmartBear Software. (2024). Retrieved from: <https://swagger.io/>
22. Snyder, K. (2023). Website Development: A Beginner's Guide. Forbes Advisor. Forbes Media LLC. Retrieved from: <https://www.forbes.com/advisor/business/website-development/>
23. Rynok rozrobky programnogo zabezpechennya. [Market of Software Development]. PRO Platforma efektyvnogo reguluvannya. Ministerstvo ekonomiky Ukrainy. PRP Platform of effective regulation. Ministry of Economics of Ukraine. (2024). Retrieved from: <https://regulation.gov.ua/dialogue/it-i-telekom/14-rinok-rozrobki-programnogo-zabezpechennya> [in Ukrainian].