УДК 004.9
DOI https://doi.org/10.32782/2663-5941/2024.1.1/32

Ляшенко О.М.
Херсонський національний технічний університет

Бодягін В.Ю.
Херсонський національний технічний університет

Кирийчук Д.Л.
Херсонський національний технічний університет

Фролова М.Е.
Херсонський національний технічний університет

ПРОЄКТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ ДЛЯ ПІДТРИМКИ ПРОЦЕСІВ КЕРУВАННЯ НАВЧАЛЬНОЮ ДІЯЛЬНІСТЮ УНІВЕРСИТЕТУ: ПІДСИСТЕМА «СТУДЕНТ»

У роботі проведено проєктування програмної системи для підтримки процесів керування навчальною діяльністю університету: підсистема «Студент».

Запропонована програмна система забезпечує обробку карток студентів, обробку інформації про успішність студентів, формування екзаменаційних відомостей студентів, формування наказів на призначення стипендії та матеріальну допомогу.

Програмна система дозволяє вводити, зберігати та аналізувати інформацію про студентів протягом усього періоду їх навчання.

Архітектура програмної системи базується на архітектурному патерні Модель-Подання-Контролер (Model-View-Controller) MVC. Патерн MVC відокремлює подання системи та взаємодію із системою від даних системи. Система поділяється на три логічних компонента, які взаємодіють один з одним. Компонент Модель керує системними даними та операціями над даними. Компонент Подання відображає дані для користувача. Компонент Контролер взаємодіє з користувачем, ініціює операції в моделі і керує роботою подання. Патерн MVC дозволяє змінювати дані незалежно від їхнього подання. Зміна даних, зроблена в одному поданні, відображається у всіх інших поданнях програмної системи.

Для розроблення програмної системи було обрано ASP.NET Web Forms, що базуються на технології Microsoft ASP.NET, у якій код, що виконується на сервері, динамічно генерує вихідні дані веб-сторінки в браузер або клієнтський пристрій. Сторінка Web Forms ASP.NET автоматично відображає правильний HTML-код, сумісний із браузером, для таких функцій, як стилі, макет тощо. Web Forms сумісні з будь-якою мовою, яку підтримує загальномовне середовище виконання .NET. Для створення Web Forms ASP.NET в роботі було використано інтегроване середовище розробки Visual Studio та мову програмування C#.

Для проєктування, моделювання та розроблення бази даних програмної системи для підтримки процесів керування навчальною діяльністю університету було використано MySQL Workbench.

Ключові слова: *програмна система, навчальна діяльність університету, ASP.NET Web Forms, мова C#.*

Постановка проблеми. Нині практично відсутній огляд сучасних теоретико-методологічних підходів до проєктування та розроблення програмних систем для підтримки процесів керування навчальною діяльністю університету, а також шляхів подальшого розвитку таких систем.

Таким чином, проєктування, моделювання та розроблення програмної системи є актуальною науково-прикладною задачею, а її побудова потребує виявлення та аналізу сучасних тенденцій розвитку та застосування таких систем, насамперед, на основі провідних наукових публікацій.

Аналіз останніх досліджень і публікацій. У роботі [1] розглянуто питання впровадження хмарних сервісів в освітній процес дисципліни «Бази даних» майбутніх учителів інформатики і проєктування на їх основі хмаро орієнтованого навчального середовища.

У роботі [2] розглянуто питання ІТ-забезпечення діяльності університету. Розглянуто структуру інтегрованої інформаційної системи університету,

методологію організації бізнес-процесів у ЗВО та їх інформаційно-аналітичну підтримку. Описано досвід розбудови IT-інфраструктури Сумського державного університету, методичні та технологічні розробки і надано відповідні рекомендації.

У роботі [3] розглянуто шляхи використання відкритого інформаційно-освітнього середовища в умовах змішаного та дистанційного навчання, його вплив на якість освітнього процесу та підготовку майбутніх фахівців в умовах цифрової трансформації вищої освіти.

У роботі [4] розроблено платформу прикладної системи для управління викладанням в коледжах та університетах, яка може допомогти персоналу ефективно обробляти різні види інформації та надавати інформаційні послуги для викладачів та студентів. Основою прикладної системи є алгоритм багаторівневої нейронної моделі.

У роботі [5] представлено концептуальну карту програмних моделей, методів і технік, а також елементів, які було використано у розробці навчальної програми. Ця концептуальна карта дозволяє створити базу знань для міждисциплінарної роботи між спеціалістами з навчальних програм, керівниками програм, викладачами та розробниками програмного забезпечення.

Метою статті є проєктування програмної системи для підтримки процесів керування навчальною діяльністю університету: підсистема «Студент».

Виклад основного матеріалу. Програмна система (ПС) забезпечує обробку карток студентів, обробку інформації про успішність студентів, формування екзаменаційних відомостей студентів, формування наказів на призначення стипендії та матеріальну допомогу.

Архітектура ПС базується на архітектурному патерні Модель-Подання-Контролер (Model-View-Controller, MVC). Патерн MVC дозволяє змінювати дані незалежно від їхнього подання. Зміна даних, зроблена в одному поданні, відображається у всіх інших поданнях ПС.

На рис. 1 патерн MVC реалізує механізм керування взаємодією з користувачем.

Після проєктування архітектури було виконано декомпозицію ПС на модулі. Фактично цю роботу можна вважати введенням у детальне проєктування, яке конкретизує архітектурні рішення. Завдання декомпозиції це завдання визначення внутрішнього змісту кожної підсистеми (модуля). Результатом її вирішення є формування структури підсистеми – набору модулів та відносин їх взаємодії.

Відомі два підходи та два типи моделей для декомпозиції підсистем на модулі: модель потоку даних та об'єктно-орієнтована модель. В роботі для декомпозиції підсистем на модулі було застосовано об'єктно-орієнтовану модель.

На рис. 2. подано приклад декомпозиції підсистеми «Студент» на модулі, що призначені для збирання та зберігання інформації про студентів.

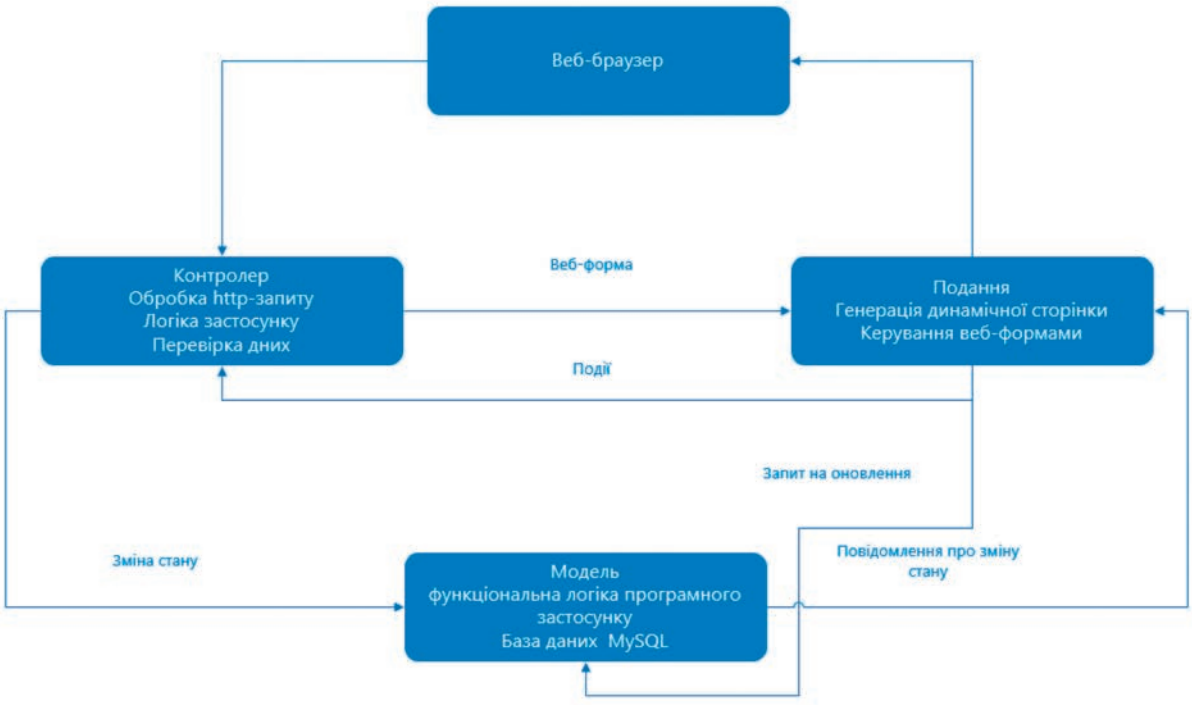

Рис. 1. Архітектура програмної системи на основі патерну MVC

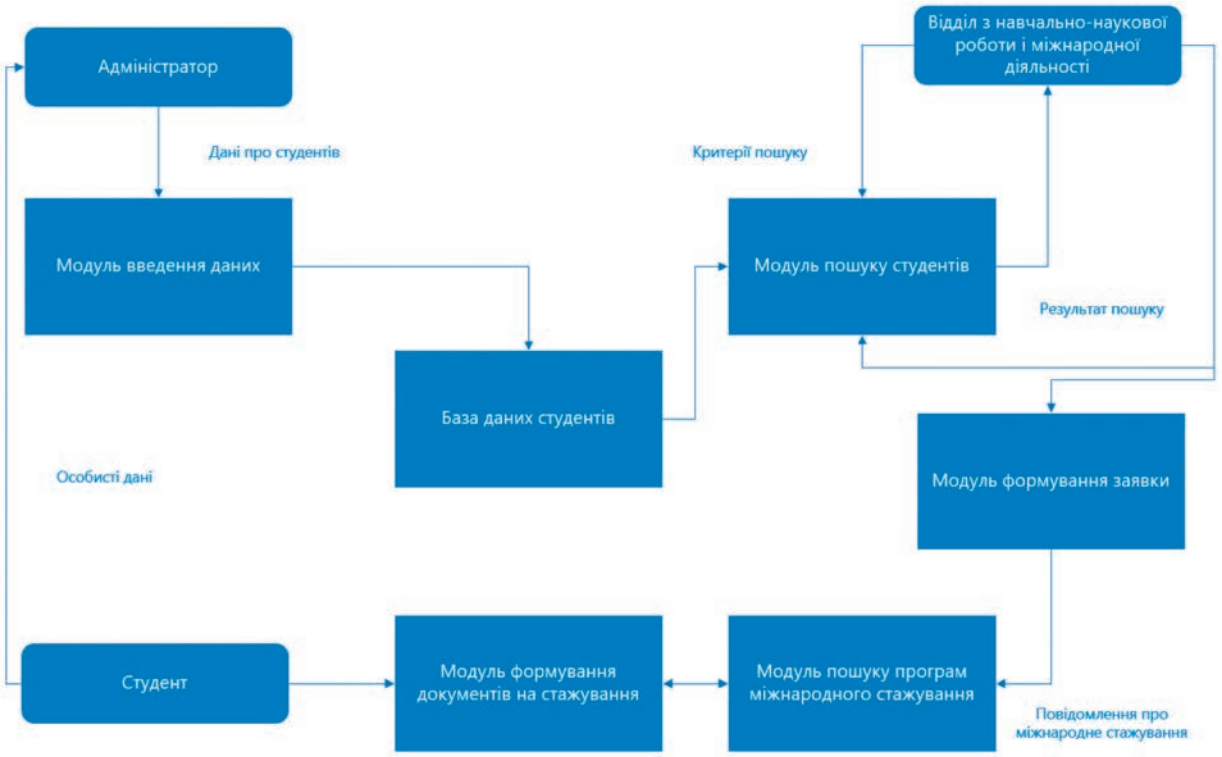

Рис. 2. Приклад декомпозиції підсистеми «Студент» на модулі

Також рис. 2 демонструє приклад процесу надання через Інтернет фахівцям Відділу з навчально-наукової роботи і міжнародної діяльності інформації про студентів. Зовнішньою сутністю є Відділ з навчально-наукової роботи і міжнародної діяльності, адміністратор та студент. Фахівець Відділу з навчально-наукової роботи і міжнародної діяльності реєструється в системі, вводить дані для пошуку кандидатів на проходження міжнародного стажування та отримує від системи результати пошуку. Адміністратор вводить інформацію про студентів, які зберігаються у базі даних. Студент може змінити свою контактну інформацію, зміни також зберігаються у базі даних. Вибраним студентам фахівець Відділу з навчально-наукової роботи і міжнародної діяльності надсилає повідомлення із пропозицією про проходження міжнародного стажування в межах програм академічної мобільності.

Для проєктування, моделювання та розроблення бази даних ПС було використано MySQL Workbench. База даних ПС містить 23 таблиці (рис. 3).

У БД виділені наступні сутності: Вчене_Звання, Місце_Народження, Громадянство, Дисципліна_Плану, Назва_Дисципліни, Рівень_Освіти, Навчальна_Картка, Екзаменаційна_Відомість, Факультет, Назва_Групи, Сімейний_Стан, Оцінка, Наказ, Місце_Проживання, Пільга, Спеціалізація, Спеціальність, Студент, Викладач, Напрямок_Підготовки, Університет.

Сутності із зв'язком 1 до N:

1. Студент – Місце_Народження.
2. Студент – Громадянство.
3. Студент – Сімейний_Стан.
4. Студент – Місце_Проживання.
5. Навчальна_Картка – Університет.
6. Навчальна_Картка – Факультет.
7. Навчальна_Картка – Рівень_Освіти.
8. Навчальна_Картка – Напрямок_Підготовки.
9. Навчальна_Картка – Спеціалізація.
10. Навчальна_Картка – Спеціальність.
11. Навчальна_Картка – Студент.
12. Навчальна_Картка – Пільга.
13. Навчальна_Картка – Наказ.

Сутності із зв'язком N до M:

1. Студент – Дисципліна_Плану.
2. Студент – Екзаменаційна_Відомість.
3. Викладач – Вчене_Звання.

Для розроблення ПС було обрано ASP.NET Web Forms, інтегроване середовище розробки Visual Studio та мову програмування C#.

Структуру ПС у середовищі Visual Studio подано на рис. 4.

У програмі є три основні сторінки: Default.aspx (Home), About.aspx і Contact.aspx. До кожної

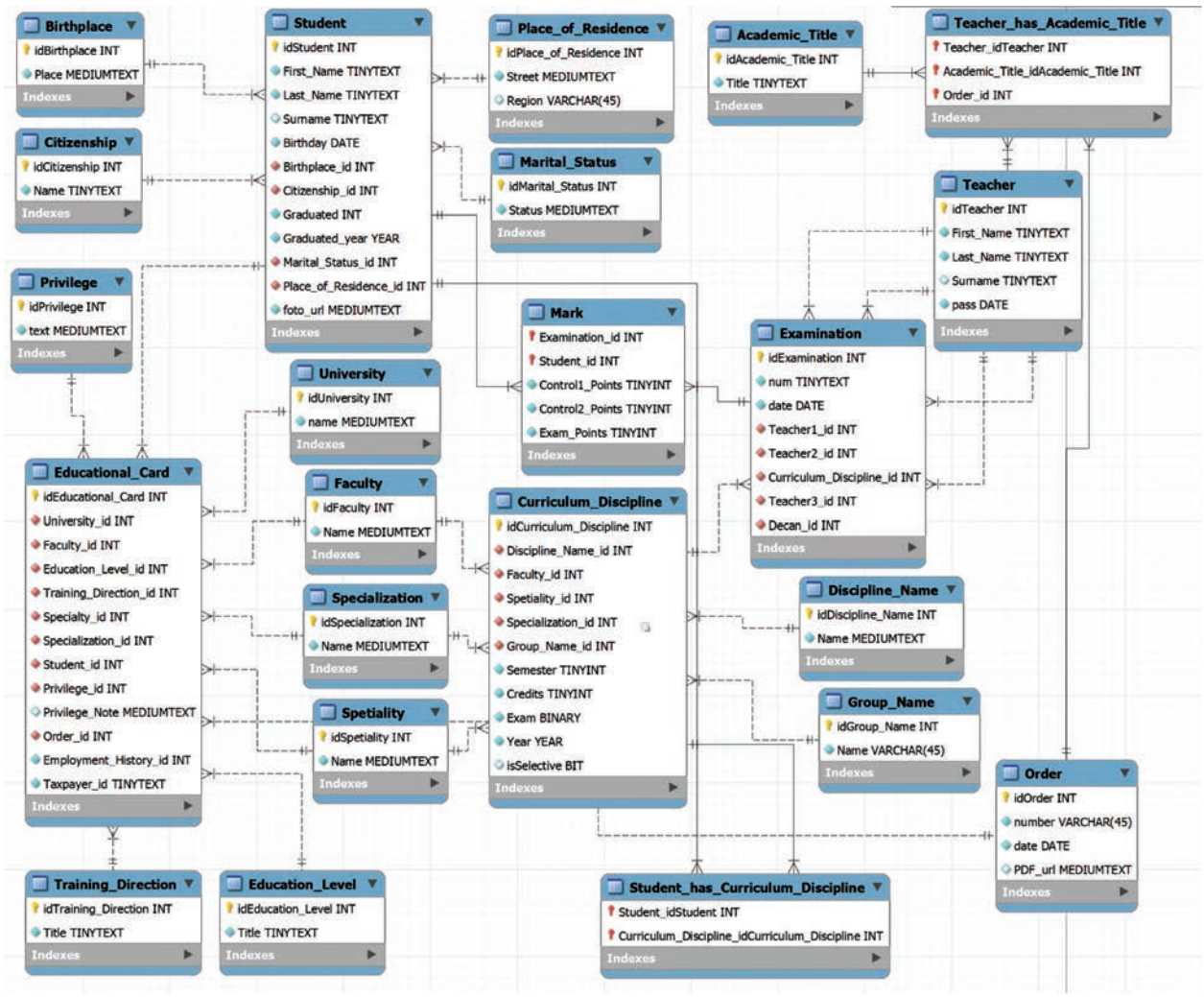

Рис. 3. Структурна схема бази даних ПС

з цих сторінок можна перейти з верхньої панелі навігації. Також є дві додаткові сторінки, які містяться в папці Account, Register.aspx і Login.aspx. Ці дві сторінки дозволяють використовувати можливості ASP.NET для створення, зберігання та перевірки облікових даних користувача [6].

Під час запуску сторінки ASP.NET Web Forms вона проходить життєвий цикл, під час якого виконується ряд кроків обробки. Ці кроки включають ініціалізацію, створення екземплярів елементів керування, відновлення та підтримку стану, запуск коду обробника подій і візуалізацію.

Коли веб-сервер отримує запит на сторінку, він знаходить сторінку, обробляє її, надсилає в браузер, а потім відкидає всю інформацію про сторінку. Якщо користувач запитує ту саму сторінку знову, сервер повторює всю послідовність, повторно обробляючи сторінку з нуля. Іншими словами, сервер не має пам'яті сторінок, які він обробив – сторінки не мають стану. Платформа сторінок ASP.NET автоматично виконує завдання підтримки стану сторінки та її елементів керування, а також надає чіткі способи підтримувати стан інформації про програму [6].

Програма включає функцію, яка реєструє користувачів і зберігає їхні облікові дані, щоб вони могли входити на веб-сайт.

ASP.NET Identity зберігає облікові дані користувачів у базі даних. Коли користувачі входять в систему, програма перевіряє їхні облікові дані, зчитуючи базу даних. Папка облікового запису проєкту містить файли, які реалізують такі функціональні можливості: реєстрація, вхід, зміна пароля та авторизація доступу. Крім того, ASP.NET Web Forms підтримує OAuth і OpenID. Ці покращення автентифікації дозволяють користувачам входити на сайт, використовуючи наявні облікові дані з таких облікових записів, як Facebook, Windows Live і Google [6].

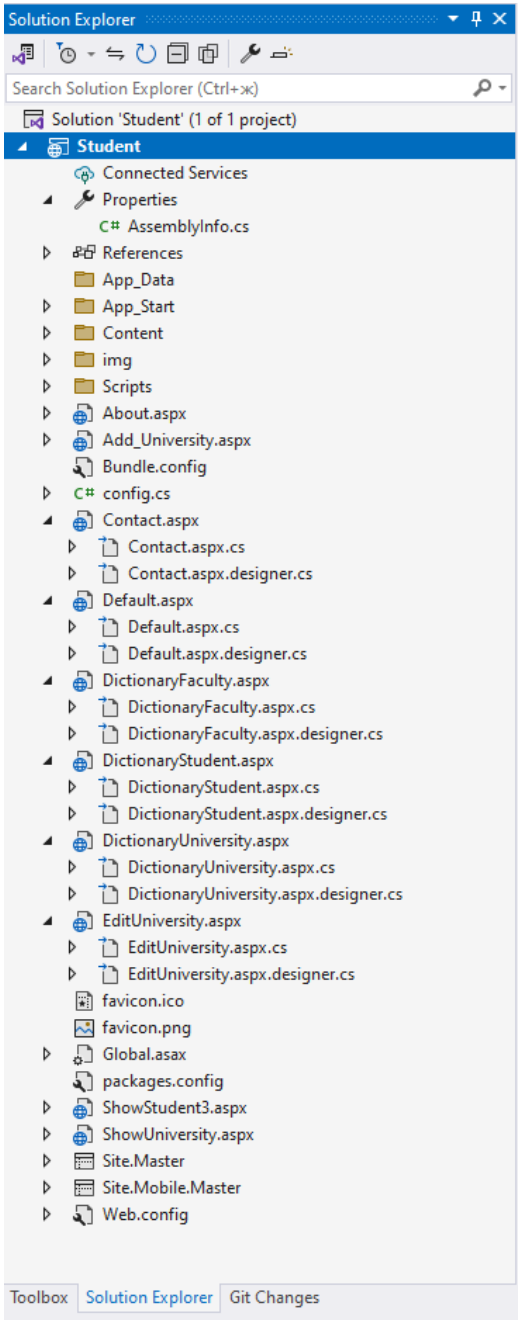

Рис. 4. Структура ПС у середовищі Visual Studio

Головна сторінка Default.aspx визначає узгоджений вигляд і поведінку для всіх сторінок у програмі.

Сторінка Site.Master дозволяє створювати узгоджений макет і використовувати стандартну поведінку для сторінок у програмі.

Global.asax – додатковий файл, який містить код для відповіді на події на рівні програми та сеансу.

Web.config – конфігураційні дані для програми.

Приклади роботи ПС подано на рис. 5–6.

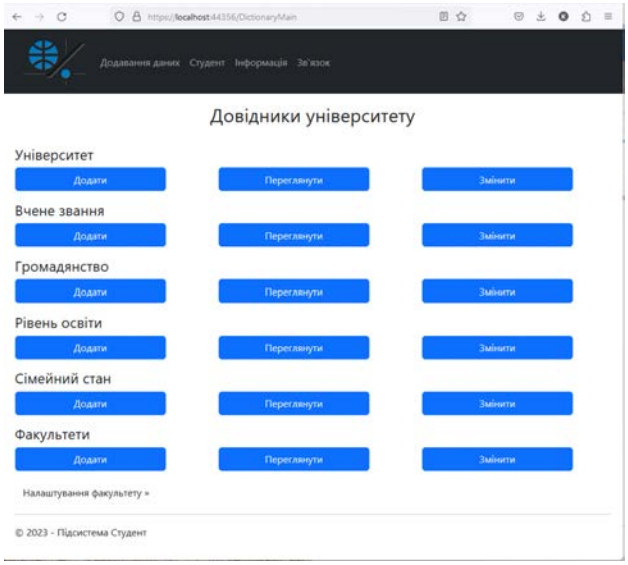

Рис. 5. Довідники університету

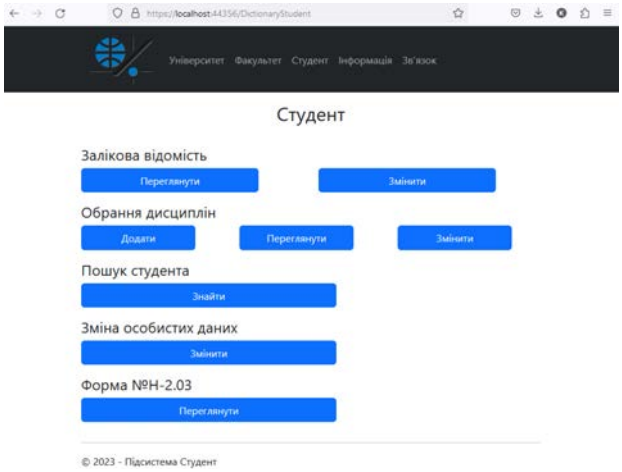

Рис. 6. Підсистема «Студент»

Висновки. У роботі описано процес архітектурного проєктування ПС. Визначено, що архітектурне проєктування дає розуміння правильної організації системи та створює структуру для цієї правильної організації, з'єднує весь етап проєктування з детальними вимогами, оскільки архітектура виділяє основні структурні компоненти системи та формує відносини між ними. Розроблено архітектуру ПС на основі патерну MVC. Описано процес розроблення структури модулів ПС. Описано елементи керування на стороні сервера, що були використані при побудові ПС. Розроблено архітектуру проєкту у середовищі Visual Studio. Наведено приклади роботи ПС.

Список літератури:

1. Кривонос О.М., Коротун О.В. Етапи проектування хмаро орієнтованого середовища навчання баз даних майбутніх учителів інформатики. *Інформаційні технології і засоби навчання*. 2018. № 1(63). С. 130–145.

2. ІТ-забезпечення діяльності інноваційного університету: досвід українського вишу: монографія/ Васильєв А.В. та ін.; за ред. проф. А.В. Васильєва. Суми: Сумський державний університет, 2016. 173 с.
3. Буйницька О. П. Система педагогічного проєктування інформаційно-освітнього середовища для здійснення підготовки майбутніх соціальних педагогів : монографія. Київ: Київський університет імені Бориса Грінченка, 2021. 568 с.
4. Ye Zi. Construction of Educational Management Software Design Pattern Based on Deep Neural Network. *Procedia Computer Science*. 2022. № 208. С. 176–183.
5. Melillan A., Cravero A., Sepulveda S. Software Development and Tool Support for Curriculum Design: A Systematic Mapping Study. *Appl. Sci*. 2023. № 13(7711). С. 2371–2390.
6. Joel Murach, Mary Delamater. Murach's ASP.NET Core MVC. Mike Murach & Associates, 2020. 780 с.

Liashenko O.M., Bodiahin V.Yu., Kyryichuk D.L., Frolova M.E. DESIGNING A SOFTWARE SYSTEM TO SUPPORT THE PROCESSES OF MANAGEMENT OF EDUCATIONAL ACTIVITIES OF THE UNIVERSITY: SUBSYSTEM "STUDENT"

The work carried out the design of a software system to support the processes of management of educational activities of the university: subsystem "Student." The proposed software system provides processing student cards, processing information about student performance, the formation of examination information of students, the formation of orders for the appointment of scholarships and material assistance. The software system allows you to enter, store and analyze information about students throughout the period of their training.

The architecture of the software system is based on the architectural pattern Model-View-Controller (MVC). The MVC pattern separates system representation and system interaction from system data. The system is divided into three logical components that interact with each other. The Model component manages system data and data operations. The View component displays data for the user. The Controller component interacts with the user, initiates operations in the model, and controls the operation of the view. MVC pattern allows you to change data regardless of its representation. A data change made in one view is reflected in all other views of the software system.

To develop a software system, ASP.NET Web Forms were chosen, based on Microsoft ASP.NET technology, in which code running on a server dynamically generates the output of a web page into a browser or client device. The Web Forms ASP.NET page automatically displays the correct browser-compatible HTML for features such as styles, layout, etc. Web Forms are compatible with any language supported by the .NET common language runtime. To create Web Forms ASP.NET, Visual Studio integrated development environment and C # programming language were used. MySQL Workbench was used to design, model and develop a software system database to support the university's learning management processes.

Key words: *software system, educational activities of the university, ASP.NET Web Forms, C # language.*