

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Одеський національний технологічний університет
Університет Інформатики і прикладних знань, м.Лодзь, Польща
Національний технічний університет України «Київський
політехнічний інститут»
Навчально-науковий інститут комп'ютерних систем і технологій
«Індустрія 4.0» ім. П.М. Платонова

XXII Всеукраїнська науково-технічна конференція
молодих вчених, аспірантів та студентів

«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ
ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»

Матеріали конференції



Одеса

21-22 квітня 2022 р.

Стан, досягнення та перспективи інформаційних систем і технологій / Матеріали XXII Всеукраїнської науково-технічної конференції молодих вчених, аспірантів та студентів. Одеса, 21-22 квітня 2022 р. - Одеса, Видавництво ОНТУ, 2022 р. – 251 с.

Збірник включає матеріали доповідей учасників конференції, які об'єднані за тематичними напрямками конференції.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова - д.т.н., проф., **Єгоров Б.В.**, ректор ОНТУ

Співголови:

Поварова Н.М. – к.т.н., доц., проректор з наукової роботи ОНТУ,
Котлик С.В. – к.т.н., доц., директор ННІКСіТ "Індустрія 4.0" ОНТУ,
Даріуш Долива, д.математичн.наук, уповноважений декана факультету Інформатики УІтаПЗ, м.Лодзь, Польща,
Ковалюк Т.В. - к.т.н., доц., Київський національний університет імені Тараса Шевченка

Члени оргкомітету:

Плотніков В. М. – д.т.н., проф., завідувач кафедри ІТтаКБ ОНТУ,
Артеменко С.В. – д.т.н., проф., завідувач кафедри КІ ОНТУ,
Хобін В.А. – д.т.н., проф., завідувач кафедри АТПтаРС ОНТУ,
Тарасенко В.П. – д.т.н., проф., завідувач кафедри СКС НТУУ «Київський політехнічний інститут»,
Невлюдов І.Ш. – д.т.н., проф., завідувач кафедри КІТАМ ХНУРЕ,
Мельник А.О. – д.т.н., проф., завідувач кафедри ЕОМ НУ “Львівська політехніка”,
Жуков І.А. – д.т.н., проф., завідувач кафедри КСтаМ НАУ.

Матеріали подано українською та англійською мовами.
Редактор збірника Котлик С.В.

ВІДПОЧИНКУ. Бондарчук О.О., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	
ОСОБЛИВОСТІ ВИКОРИСТАННЯ ІНТЕРНЕТ ПРОТОКОЛІВ ДЛЯ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ПОШТОЮ. Веренько А.І., Романюк О.В. (Вінницький національний технічний університет)	108
СИСТЕМА ОБЛІКУ СТУДЕНТІВ КАФЕДРИ. Власов Р.І., Свинчук О.В., Євтушенко А.М. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	110
ІМПЛЕМЕНТАЦІЯ КОМПОНЕНТІВ СИНТЕЗУ ТА АНАЛІЗУ МУЗИЧНИХ ЗВУКІВ. Войтко В.В., Бевз С.В., Бурбело С.М., Ставицький П.В. (Вінницький національний технічний університет)	112
ЗАГАЛЬНІ ПІДХОДИ ДО РОЗРОБКИ ERP-СИСТЕМ, ЩО ІНТЕГРУЮТЬ E-COMMERCE СИСТЕМИ. Войтко В.В., Позур М.Ю., Денисюк А.В. (Вінницький національний технічний університет)	113
РОЗРОБКА ДОДАТКУ ДЛЯ АВТОМАТИЗАЦІЇ ЗАМОВЛЕННЯ ДОСТАВКИ З РЕСТОРАНУ. Гарас С.Я. (Фаховий коледж промислової автоматики та інформаційних технологій ОНАХТ)	115
ІНТЕРАКТИВНИЙ ВЕБ-САЙТ КАФЕДРИ. Глушенко І.С., Бандурка О.І., Свинчук О.В. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	117
ВИКОРИСТАННЯ ЛІНГВІСТИЧНИХ ЗМІННИХ В ОЦІНЮВАННІ ТЕСТУВАННЯ. Головня Д. М., Лютенко І. В. (Національний технічний університет «Харківський політехнічний інститут»)	119
ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ОБЛІКУ ПРОВЕДЕНОГО ЧАСУ ЗА КОМП'ЮТЕРОМ. Дорошенко А.С., Снігур Т.С. (Одеський національний технологічний університет)	121
КЛІЄНТ-СЕРВЕРНИЙ ДОДАТОК ДЛЯ КОМУНІКАЦІЇ ПО ЛОКАЛЬНІЙ МЕРЕЖІ. Єременко К.Х., Бандурка О.І., Свинчук О.В. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	122
СУЧАСНИЙ ПІДХІД ДО ПРОЕКТУВАННЯ ПРОГРАМНОЇ АРХІТЕКТУРИ FULL – STACK ДОДАТКІВ. Жадан А.С., Селіванова А.В. (Одеський національний технологічний університет)	124
ІНФОРМАЦІЙНА СИСТЕМА ПОШУК РЕПЕТИТОРА НА БАЗІ СЕРВЕРА WAMP. Здробилко Н.Ю. Здолбіцька Н.В. (Луцький національний технічний університет)	126
ОПТИМІЗАЦІЯ РОБОТИ КОМП'ЮТЕРА ЗА ДОПОМОГОЮ ЖЕСТІВ З ВИКОРИСТАННЯМ ARDUINO. Ісайко С.В. (Фаховий коледж промислової автоматики та інформаційних технологій ОНАХТ)	128
ІНТЕРНЕТ-МАГАЗИН З ПРОДАЖУ ВЗУТТЯ. Каковкіна К.І., Швець Н.В. (Одеський національний технологічний університет)	130
ПРОГРАМНА ПІДТРИМКА НАВЧАННЯ АЛГОРИТМІВ СОРТУВАННЯ ОДНОРІДНИХ ДАНИХ. Карелін М., Черненко В. (Вище професійне училище №7 м. Кременчука)	131
ПОРІВНЯННЯ ЕФЕКТИВНОСТІ СЕРЕДОВИЩ ДЛЯ АВТОМАТИЗАЦІЇ З МАНУАЛЬНИМ МЕТОДОМ ПІД ЧАС ТЕСТУВАННЯ ВЕБ-ДОДАТКІВ. Клестова Д.М., Гришанович Т.О. (Волинський національний університет імені Лесі Українки)	133
ОСОБЛИВОСТІ ВИКОРИСТАННЯ ТЕХНОЛОГІЇ AIRFLOW ДЛЯ МОНІТОРИГУ ТА ПЛАНУВАННЯ РОБОЧИХ ПРОЦЕСІВ. Ковтун Б.В., Романюк О.В. (Вінницький національний технічний університет)	135
МОБІЛЬНЕ ЗАСТОСУВАННЯ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ МІКРОНАВЧАННЯ. Комлева Н.О., М'яснікова К.О., Мельник Д.А. (Державний університет «Одеська політехніка»)	137



Рисунок 2 – Діаграма прецедентів поточного додатка

На рис.2 зображено діаграму прецедентів, яка показує дії користувача в програмі. Кожна кнопка викликає свій модуль, який працює з поставленою задачею.

Розроблена програма дозволить користувачам вільно спілкуватися в корпоративному середовищі, що значно спростить надходження потоку необхідної інформації та швидкого вирішення виробничих питань у повсякденному житті. Дана система передбачає не лише зручність, але й не високу собівартість порівняно з іншими дорого вартісними програмними додатками.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. How sockets work [Електронний ресурс]. Режим доступу: <https://www.ibm.com/docs/en/i/7.1?topic=programming-how-sockets-work>
2. Remote development made easy [Електронний ресурс]. Режим доступу: <https://towardsdatascience.com/remote-development-with-pycharm-d741287e07de?gi=51b8db3978c2>

УДК 004.4: 338.32.053.4

СУЧАСНИЙ ПІДХІД ДО ПРОЕКТУВАННЯ ПРОГРАМНОЇ АРХІТЕКТУРИ FULL – STACK ДОДАТКІВ

ЖАДАН А. С., СЕЛІВАНОВА А. В.
(arthur.zhadan@gmail.com, av_selivanova@ukr.net)
Одеський національний технологічний університет

Для покращення масштабованості програмних додатків потрібен гнучкий підхід до проектування архітектури цих додатків. Розглянутий підхід допоможе покращити таку властивість як масштабування у програмних додатків. Об'єктом дослідження є процес покращення аспекту масштабованості при розробці програмних додатків за допомогою гнучкого підходу у проектуванні архітектури. Предметом дослідження є відстеження сучасних тенденцій у виборі програмної архітектури у розробці програмних додатків.

В архітектурі програмного забезпечення масштабованість - це здатність підтримувати велику кількість архітектурних компонентів та з'єднань між ними.

Програмна архітектура – це спосіб структурування програмної або обчислювальної системи, абстракція елементів системи на певній фазі її роботи. Система може складатись з кількох рівнів абстракції і мати багато фаз роботи, кожна з яких може мати окрему архітектуру.

Система може складатись з кількох рівнів абстракції і мати багато фаз роботи, кожна з яких може мати окрему архітектуру.

Додатки, котрі використовують доступ до інтернету, частіше всього являються клієнт – серверними.

WEB – сервер – це сервер, що приймає HTTP – запити від клієнтів, зазвичай веб-браузерів, і видає HTTP-відповіді.

Клієнт – це додаток, що відправляє HTTP – запити до WEB – серверу.

HTTP – це протокол передачі даних, що використовується в комп'ютерних мережах. Він підтримує такі операції як GET, POST, PUT, DELETE. Операції цього протоколу реалізують CRUD операції у додатках.

В свою чергу, клієнт–серверні додатки можуть мати монолітну або відокремлену архітектуру.

Монолітна архітектура – це такий спосіб організації програмних модулів, при якому серверний та програмний код представляють собою один єдиний додаток. Зазвичай, монолітну архітектуру не використовують при розробці додатків, котрі мають клієнти під різні операційні системи, тобто вона є менш гнучкою.

Відокремлена архітектура – це такий спосіб організації програмних модулів, при якому серверний та клієнтський код представляють собою два окремих додатків. Відокремлена архітектура є більш гнучкою, так як дозволяє відокремити серверну частину від клієнтської. При такому відокремленні часто використовуються таке поняття як API.

API – це програмний інтерфейс, котрий описує методи, якими одна комп'ютерна програма може взаємодіяти з іншою програмою. Прикладом відокремленою архітектури може REST API.

REST API (Representational State Transfer Application Interface) – це такий архітектурний стиль, при якому серверна частина спілкується з клієнтською частиною та навпаки завдяки протоколу HTTP та представляє данні у JSON або YAML форматі.

Важливо також зауважити, що при розробці програмного забезпечення, котре будується на відокремленій архітектурі також використовуються такі поняття як Back – End, Front – End та Full – Stack.

Back – End розробка – це розробка серверної частини додатку. Front – End розробка – це розробка клієнтської частини додатку. Full – Stack розробка – це одночасна розробка серверної (Back – End) та клієнтської (Front – End) частин додатку, зазвичай одною людиною.

Крім того, на сьогоднішній день при розробці програмного забезпечення використовуються фреймворки, бібліотеки та шаблони проектування.

Фреймворк – це програмний каркас, що полегшує розробку та об'єднання різних компонентів великого програмного проекту.

Бібліотека – це збірка об'єктів чи підпрограм для вирішення близьких за тематикою задач.

Шаблони проектування – це ефектні способи вирішення задач проектування програмного забезпечення.

Данні засоби допомагають писати підтримуваний код (чистий та структурований), особливо це важливо, коли над проектом працює команда розробників.

Зазвичай кожен фреймворк включає бібліотеки та реалізує шаблони проектування.

Розглянемо популярні архітектурні шаблони проектування:

- Model-View-Controller (MVC) або Model-View-Presenter (MVP);

- ModelView-ViewModel (MVVM).

MVC та MVP дуже схожі, модель в даному випадку – це бізнес логіка додатку (наприклад, сутності бази даних), вид – це XML подібна розмітка графічного інтерфейсу додатку або JSON дані, контролер або презентер їх зв'язує та поновлює. Зазвичай використовується для серверної частини в відокремленій архітектурі або для монолітних додатків. Прикладом є фреймворк ASP.NET, Django, Laravel [1] [2] [3].

MVVM використовується в відокремленій архітектурі для клієнтських додатків. Він зчитує JSON дані від серверу та відправляє їх до клієнтського графічного інтерфейсу і зазвичай це все робиться в одному файлі. Прикладом можуть бути реактивні JavaScript фреймворки, наприклад VueJS [1] [2] [3].

Таким чином, враховуючі останні тенденції у розробці програмного забезпечення, доцільно використовувати відокремлену архітектуру при розробці середніх або великих додатків, котра складається з серверної (Back – End) та клієнтської (Front – End) частин. Обидві ці частини можна зв'язати між собою завдяки програмному інтерфейсу (REST API). Це дасть можливість у майбутньому розробляти клієнти під різні операційні системи. Крім того, при розробці доцільно використовувати фреймворки, бібліотеки та шаблони проектування з метою написання чистого, підтримуваного коду та продуктивної розробки. Для розробки маленьких додатків доцільно використовувати монолітну архітектуру, так як вона більш простіша і потребує менше затрат. Потрібно пам'ятати, що відокремлена архітектура потребує більше часу та сил на реалізацію, але дає гарну масштабованість, в свою чергу монолітна архітектура більш проста та швидка у реалізації.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. ASP.NET Documentation: [Веб-сайт]. URL: <https://www.django-rest-framework.org/> (дата звернення: 15.04.2022).
2. Django REST Framework Documentation: [Веб-сайт]. URL: <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-6.0/> (дата звернення: 15.04.2022).
3. Laravel Documentation: [Веб-сайт]. URL: <https://laravel.com/docs/9.x/> (дата звернення: 15.04.2022).

УДК 004.42

ІНФОРМАЦІЙНА СИСТЕМА ПОШУК РЕПЕТИТОРА НА БАЗІ СЕРВЕРА WAMP

ЗДОЛБИЦЬКА Н.В. (ninazdolb@gmail.com),
ЗДРОБИЛКО Н.Ю. (nazik.zdrobilko@gmail.com)
Луцький національний технічний університет (Україна)

Розглядається реалізація інформаційної системи на базі сервера WAMP, на мові PHP та з використанням MySQL, для пошуку репетиторів, розглянуто різні системи-аналоги.

Постановка проблеми в загальному вигляді

Щорічно пошук репетиторів стає все більш актуальним. Все більше студентів та школярів шукають собі репетиторів для покращення своїх знань, оскільки дуже багато викладачів старшого покоління, використовують застарілі методи навчання, саме через це розробляється багато програмних засобів для швидкого пошуку репетитора. Сфера репетиторства розвивається з кожним роком все більше, через епідемічну ситуацію у світі все популярнішими стають індивідуальні та групові онлайн заняття.

Метою даної роботи є розробка програмного проекту для пошуку репетиторів з різноманітних дисциплін та для підготовки школярів, студентів до різних видів перевірки

**XXII Всеукраїнська науково-технічна конференція
молодих вчених, аспірантів та студентів**

**«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ
ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»**

Одеса

21-22 квітня 2022 р

Збірник включає доповіді учасників конференції. Тези доповідей публікуються у вигляді, в якому вони були подані авторами.

Відповідальність за зміст і форму подачі матеріалу несуть автори статей.

Редакційна колегія: Котлик С.В., Корнієнко Ю.К.

Комп'ютерний набір і верстка: Соколова О.П.

Відповідальний за випуск: Котлик С.В.