



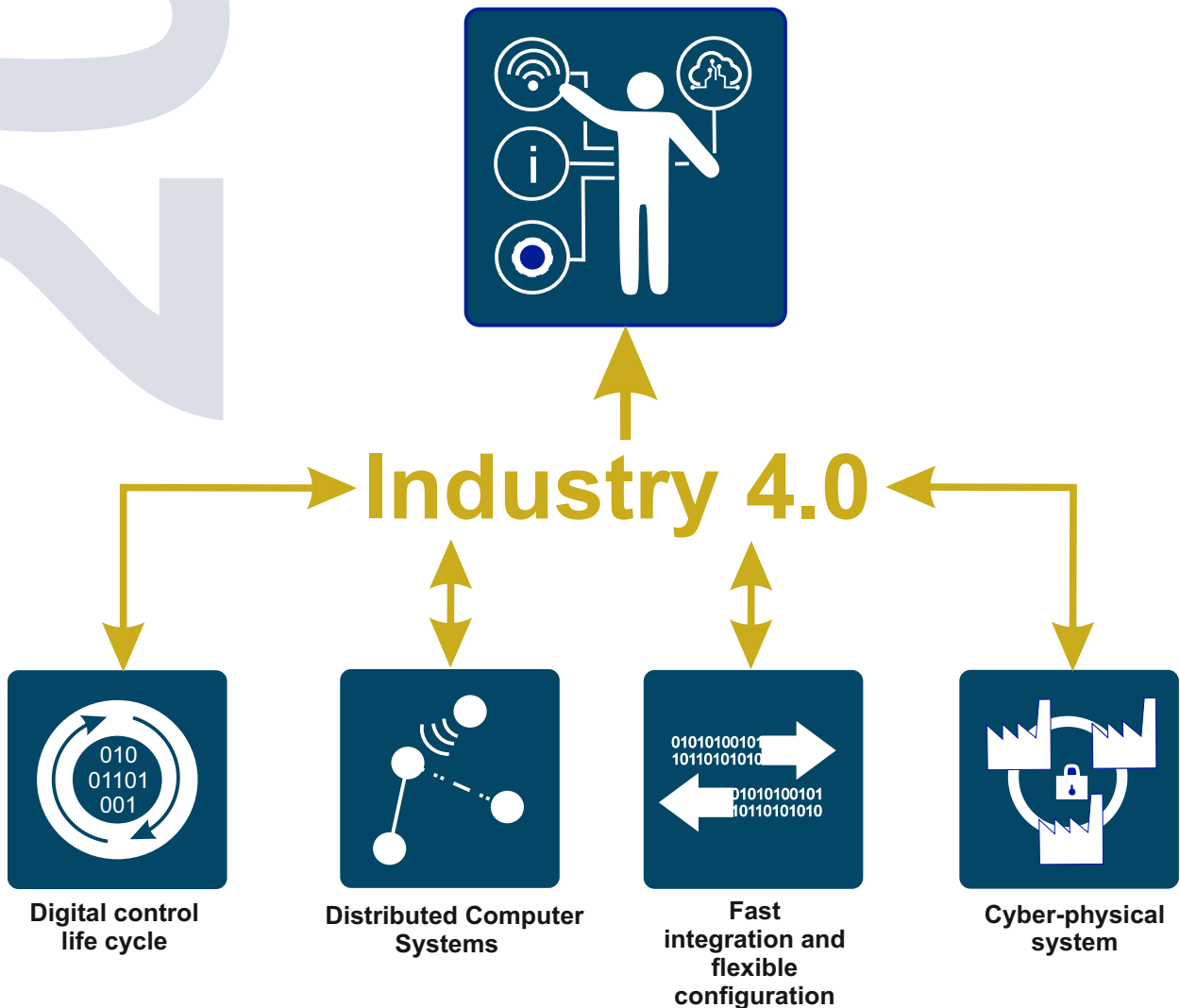
The Ministry of  
Education and Science  
of Ukraine

<https://nure.ua/>

Kharkiv National  
University of  
Radio Electronics

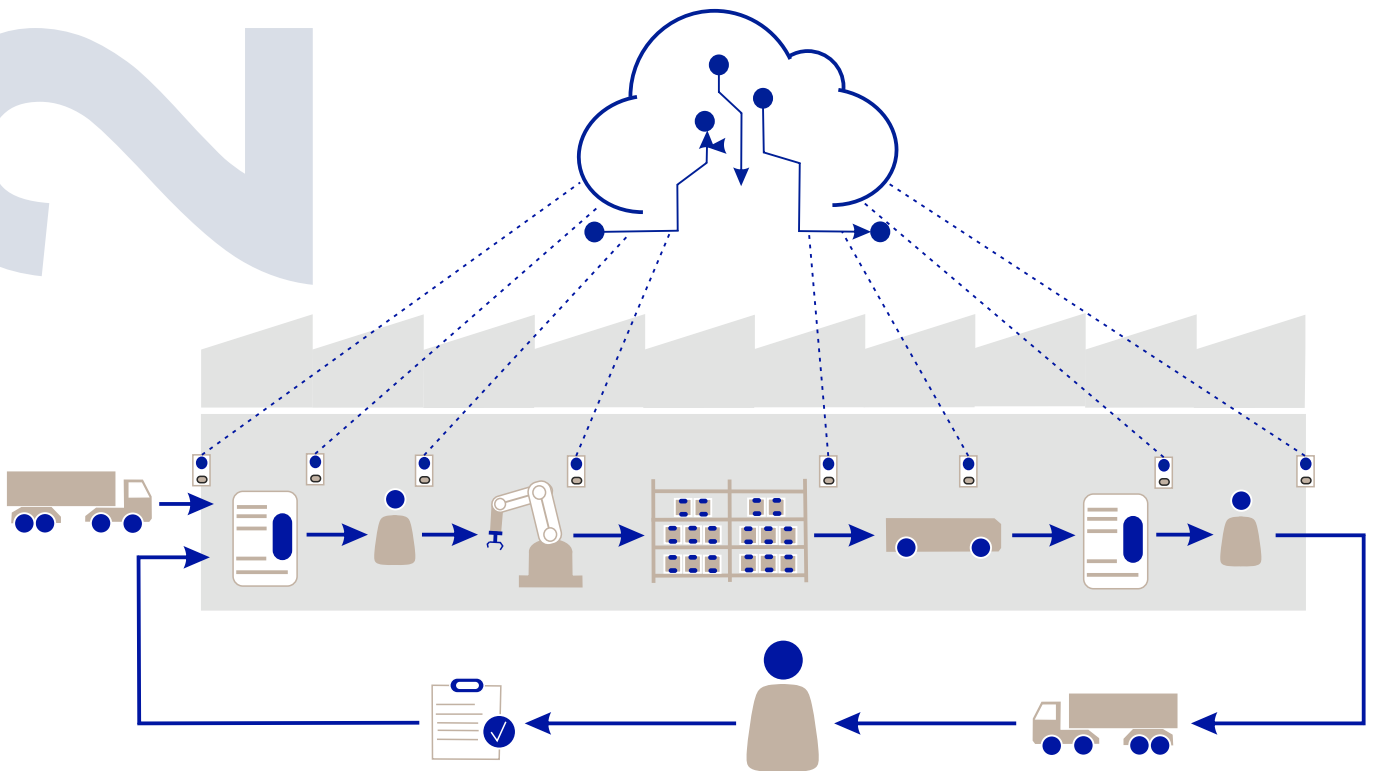
**KITAM**

**COLLECTION**  
OF STUDENTS' SCIENTIFIC PAPER  
«Automation and Development of Electronic Devices»  
ADED-2022  
(Part 2)



# ЗБІРНИК

студентських наукових статей  
«Автоматизація та приладобудування»  
ADED-2022  
(Випуск 2)  
[електронне видання]



→ Industry 4.0

- Головий редактор** **Невлюдов Ігор Шакирович**, доктор технічних наук, професор, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харківського національного університету радіоелектроніки.
- Редакційна колегія:** **Филипенко Олександр Іванович**, доктор технічних наук, професор, декан факультету Автоматики та комп'ютеризованих технологій, Харківського національного університету радіоелектроніки.  
**Цимбал Олександр Михайлович**, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харківського національного університету радіоелектроніки.  
**Андрусевич Анатолій Олександрович**, доктор технічних наук, професор, начальник Криворізького коледжу національного авіаційного університету  
**Косенко Віктор Васильович**, доктор технічних наук, професор, зам. директора Державного підприємство «Південний державний проектно-конструкторський та науково-дослідний інститут авіаційної промисловості».  
**Замірець Микола Васильович**, доктор технічних наук, професор, директор Державного підприємства Науково-дослідного технологічного інституту приладобудування.  
**Свищ Володимир Митрофанович**, доктор технічних наук, професор, радник директора Державне науково-виробниче підприємство «Об'єднання Комунар».  
**Фомовська Олена Владиславівна**, кандидат технічних наук, доцент завідувач кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського.  
**Кухаренко Дмитро Володимирович**, кандидат технічних наук, доцент кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського  
**Демська Наталія Павлівна**, кандидат технічних наук, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харківського національного університету радіоелектроніки.  
**Фурманова Наталія Іванівна**, кандидат технічних наук, доцент, в.о. декана факультета Радіоелектроніки і телекомунікацій, Національного університету «Запорізька політехніка».
- Відповідальний редактор:** **Євсєєв Владислав В'ячеславович**, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харківського національного університету радіоелектроніки.

## АНАЛІЗ АРХІТЕКТУРИ СУЧАСНИХ ВЕБ-ДОДАТКІВ

**О. Г. Боков**

Харківський національний університет радіоелектроніки  
Україна, 61166, Харків, пр. Науки 14  
E-mail: oleksandr.bokov@nure.ua

**Анотація:** У даній статті розглянуто та проаналізовані архітектури сучасних веб додатків. В результаті визначено характеристики, переваги та недоліки, що допомагають вирішити що використовувати для будь-якого проекту. Аналіз формує уявлення про застосування різних архітектур відповідно до вимог та обмежень комерційного проекту під час використання та розробки.

**Ключові слова:** веб, архітектура, моноліт, мікросервіси.

## ANALYSIS OF MODERN MOBILE PLATFORMS OF GROUND SEARCH ROBOTS

**O. Bokov**

Kharkiv National University of Radio Electronics  
Ukraine, 61166, Kharkiv, Nauky av.,14  
E-mail: oleksandr.bokov@nure.ua

**Annotation:** This article considered and analyzed the architectures of modern web applications. As a result, the characteristics, advantages and disadvantages are determined, which help to decide what to use for any project. The analysis provides insight into the application of different architectures according to the requirements and constraints of the commercial project during use and development.

**Key words:** web, architecture, monolit, microservices.

**АКТУАЛЬНІСТЬ РОБОТИ.** Актуальність аналізу відмінностей у конструкції архітектури веб застосунків, що можуть використатися як дизайн майбутнього комерційного продукту. Резон використання при тих чи інших обставинах. Супутні технології та підводні каміння майбутнього вибору.

**МАТЕРІАЛ І РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ.** В якості архітектур обрані дві найрозповсюджені архітектури:

- Моноліт;
- Мікросервіси;

Розглянемо моноліт, це типічний веб застосунок зазвичай написаний на одній мові програмування, також код усього проекту зберігається у одному git репозиторію. Над ним може працювати лише одна команда.

В ході проведеного аналізу визначені наступні переваги розглянутої архітектури:

- Простота розробки;
- Мінімальні зусилля на побудові архітектури;
- найдешевший спосіб протестувати бізнес гіпотезу;
- Простота взаємодії бізнес сутностей у застосунку;
- Дуже простий процес деплою на сервера;

Недоліки:

- Чим більша команда розробників проекту тим менша продуктивність кожного з учасників;
- Проблеми з горизонтальним ростом, тобто підчас великого навантаження фінансово складно її витримати;
- Купа легасі коду, з котрим потрібно щось робити (написаного ще під час тестування бізнес гепотізи);

- Ніяка відмово витривалість, якщо щось впало то увесь застосунок рахується впавшим;

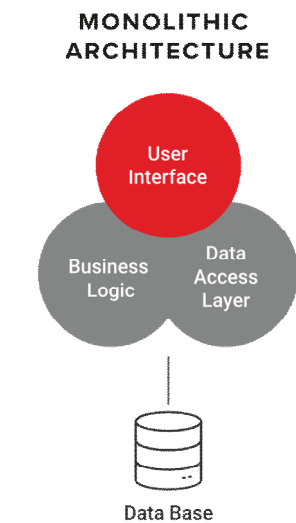


Рисунок 1 – Моноліт [1]

А тепер розглянемо реальний кейс. Instagram.

Проект як і більшість починався з перевірки бізнес ідей, це логічно бо ніхто навіть не міг подумати що додаток стане таким популярним який, по факту, мав лише дві функціональності: постити фотки, ставити лайки. І це все при реальних конкурентах, таких як Facebook та Twitter (вони реалізовували ці ж функціональності). Але це все лірика перейдемо до технологій)

Це повний дефолт для стартапу чи перевірки бізнес ідеї, вони використовували PostgreSQL як єдина БД (потім додалася Cassandra), Python3 як основна мова програмування додатку та потужний фреймворк Django (не с точки зору продуктивності кінцевого додатку, а швидкості розробки).

Після того як стартап вистрелив вони зустріли усі описані вище проблеми. Але вони вирішили миритися з ними та не перестроюватися на іншу архітектуру.

Розглянемо мікросервісну архітектуру, чудово підходить для складних високонавантажених проектів, код мікросервісів зазвичай зберігається у окремих git репозиторіях, над кожним мікросервісом працюють до 10 девелоперів.[2]

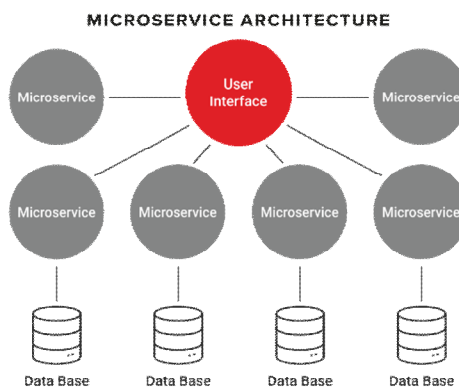


Рисунок 2 – Мікросервісна архітектура [3]

В ході проведеного аналізу визначені наступні переваги розглянутої архітектури:

- Висока продуктивність кожного з учасників проекту;
- Мікросервіси не залежать один від одного;
- Ефективне горизонтальне масштабування;

- Відносна відмова витривалість;
- Комфортне локальне тестування кожного з мікросервісів;

Недоліки:

- Нереальні зусилля для побудови архітектури;
- Підходить тільки для перспективних проєктів;
- Транспортні накладки під час спілкування різних мікросервісів;
- Окрема БД для кожного з мікросервісів;
- Складний деплой на сервера;
- Труднощі при логуванні запитів;

А тепер розглянемо реальний кейс. Проєкт над яким я працював.

Це онлайн гра - футбол, у якості футбольних гравців використовувалися генеровані NFT картки з унікальними характеристиками. Генерація карток відбувалася за єдину суму етеріуму. Матчі відбувалися у режимі реального часу, між двома клієнтами. Клієнти мали змогу робити ставки на свої матчі та торгувати картками. Власником проєкта є впевнена у собі компанія по розробці ігор, тому вирішили вкластися у нього наповну.

Обрані мікросервіси:

- Основний ігровий сервіс;
- Сервіс авторизації;
- Сервіс мінтингу NFT карток;
- Маркетплейс сервіс;
- Кілька сервісів для інтеграції з крипто кошильками;

Боротьба з недоліками архітектури.

Транспортні накладки під час спілкування різних мікросервісів - вирішилися завдяки хорошому дизайну та побудовою кожного з мікросервісів під окрему частину бізнес логіки, а допомогло майже повністю уникнути спілкування між ними.

Труднощі при логуванні запитів - також вирішив дизайн, та окреме сховище для логів по типу Graylog.

**ВИСНОВКИ.** На основі аналізу доступної інформації щодо переваг та недоліків архітектур можна зробити висновки:

Монолітна архітектура чудово підходить для невеликих проєктів з невеликим навантаженням, та стартапів.

Мікросервісна для великих та високонавантажених проєктів.

## ЛІТЕРАТУРА

1. Monolithic vs. Microservices Architecture, Режим доступу: <https://articles.microservices.com/monolithic-vs-microservices-architecture-5c4848858f59>. – (дата звернення: 02.11.2022).
2. Microservices, Режим доступу: <https://www.chakray.com/expertise/microservices/>. – (дата звернення: 03.11.2022).
3. Architectural Design Patterns: Microservices Architecture, Режим доступу: <https://www.encora.com/insights/architectural-design-patterns-microservices-architecture>. – (дата звернення: 10.11.2022).
4. Nevliudov, I., & et al. (2021). GUI Elements and Windows Form Formalization Parameters and Events Method to Automate the Process of Additive CyberDesign CPPS Development. *Advances in Dynamical Systems and Applications*, 16(2), 441-455.