

**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**

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

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

*Матеріали конференції*



Одеса

**20-21 квітня 2023 р.**

Стан, досягнення та перспективи інформаційних систем і технологій / Матеріали XXIII Всеукраїнської науково-технічної конференції молодих вчених, аспірантів та студентів. Одеса, 20-21 квітня 2023 р. - Одеса, Видавництво ОНТУ, 2023 р. – 449 с.

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

Збірник буде корисним як для фахівців і працівників фірм, зайнятих в області ІТ, так і для викладачів, магістрів і студентів вищих навчальних закладів, які навчаються за напрямками і спеціальностями програмного забезпечення, обчислювальної техніки і автоматизованих систем, прикладної математики та обробки інформації, буде корисним професіоналам з комп'ютерного моделювання та розробки комп'ютерних ігор.

Результати досліджень у збірнику представляють собою своєрідний зріз сучасного стану справ в перерахованих галузях знань, який може допомогти як фахівцям, так і студентам університетів скласти загальну картину розвитку інформаційних технологій та пов'язаних з ними питань.

Наукові праці згруповані за напрямками роботи конференції та наведені в алфавітному порядку прізвищ авторів.

Матеріали (тези доповідей) друкуються в авторській редакції. Відповідальність за якість та зміст публікацій несе автор.

Матеріали подано українською та англійською мовами.

Редактор збірника Котлик С.В.

## **ПРЕЗИДІЯ ТА ОРГКОМІТЕТ КОНФЕРЕНЦІЇ**

### **ГОЛОВА ПРЕЗИДІЇ**

**Єгоров Б.В.**, Президент ОНТУ, академік НААН України, д.т.н., професор

### **ЧЛЕНИ ПРЕЗИДІЇ**

**Іванченкова Л.В.**, Ректор Одеського національного технологічного університету, д.е.н., професор

**Поварова Н.М.**, проректор з наукової роботи, к.т.н., доцент

**Даріуш Долива**, уповноважений декана факультету Інформатики УІтаПЗ, м.Лодзь, д.математичн.наук, Польща

**Ковалюк Т.В.** - к.т.н., доц., Київський національний університет імені Тараса Шевченка

### **ГОЛОВА ОРГКОМІТЕТУ**

**Котлик С.В.** – директор ННІКСіТ "Індустрія 4.0" ОНТУ, к.т.н., доц.

### **ЗАСТУПНИК ГОЛОВИ ОРГКОМІТЕТУ**

**Артеменко С.В.** – завідувач кафедри КІ ОНТУ, д.т.н., проф.

### **ЧЛЕНИ ОРГКОМІТЕТУ**

**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.

47. Застосування доповненої реальності для проектування інтерактивного інтерфейсу користувача. <b>Павлова О.О., Башта А.Р.</b> (Хмельницький національний університет )	263
48. Розробка інформаційної системи для підтримки навчально-виховного процесу у дошкільних навчальних закладах. <b>Подлінова М.А., Котлик С.В.</b> (Одеський національний технологічний університет)	265
49. Використання рефлексії в процесі розробки додатків на платформі .NET. <b>Позур М.Ю., Войтко В.В.</b> (Вінницький національний технічний університет)	267
50. Принципи проектування мобільного додатку. <b>Попова В.Р., Сахарова С.В.</b> (Одеський національний технологічний університет)	269
51. Інформаційна система sm, як один з перспективних напрямків розвитку мікросервісної архітектури. <b>Пригода А. Я.</b> (Державний торговельно-економічний університет)	270
52. WEB-додаток «Твій день». <b>Прокопова А.Ю., Снігур Т.С.</b> (Одеський національний технологічний університет)	272
53. Історія формування технологій проектування інформаційних систем. <b>Рашевська Н. В., Велієва В.О.</b> (Державний біотехнологічний університет)	274
54. Компютерна програма для тренування в операторів безпілотних літальних апаратів звукової ідентифікації об'єктів. <b>Романюк О. Н., Захарчук М. Д., Кулешов В.В., Шевчук Р.П., Романюк О.В.</b> (Вінницький національний технічний університет)	276
55. Розробка мобільного застосунку для пошуку тлумачення біологічних термінів. <b>Сердюк А.С., Кательніков Д.І.</b> (Вінницький національний технічний університет)	278
56. Розробка мобільного програмного застосунку з Flutter. <b>Склярів Л.С., Ломовцев П.Б.</b> (Одеський національний технологічний університет)	280
57. Можливості спільного використання PHP та XML для обміну даними. <b>Слушна Н.В.</b> (Одеський національний технологічний університет)	281
58. Features of automatic assessment of side natural illumination of premises. <b>Соколан Ю.С., Майдан П.С.</b> (Хмельницький національний університет)	282
59. Використання синтетичних компонентів як шаблонів для генерування коду. <b>Ставицький П.В., Войтко В.В.</b> (Вінницький національний технічний університет)	284
60. Дослідження технологій Blockchain для створення захищених платформ онлайн голосування. <b>Федчун К. Ю.</b> (Волинський національний університет імені Лесі Українки)	286
61. Досвід створення сучасних програмних додатків на кафедрі програмного забезпечення Вінницького національного технічного університету. <b>Хошаба О.М.</b> (Вінницький національний технічний університет)	287
62. Розроблення модуля «Робота фотостудії» на базі Веб-технологій. <b>Чернишенко М. Д.</b> (Харківський національний економічний університет імені Семена Кузнеця)	289
63. Використання графіки у браузері та її вплив на якість Веб-застосунку. <b>Чернявський М.О., Селіванова А.В.</b> (Одеський національний технологічний університет)	291
64. Концепція інтерфейсу користувача системи підтримки прийняття рішень для вибору виду спорту на основі морфофункціональних показників людини. <b>Швайко В.К., Ільчишина Ю.В., Павлова О.О.</b> (Хмельницький національний університет)	293
65. Програмне забезпечення для автоматизації роботи автосервісу. <b>Шип Д. В., Швець Н. В.</b> (ВСП "Фаховий коледж промислової автоматики та інформаційних технологій ОНТУ)	295
66. Інтернет речей - стратегічний інструмент розвитку інформаційних технологій. <b>Юскович-Жуковська В.І., Лотюк Ю.Г., Соловей Л.Я.</b> (ПВНЗ «Міжнародний економіко-гуманітарний університет ім. академіка Степана Дем'янчука»)	296
67. Дослідження особливостей тестування знань на основі розробленого додатку.	298

We compared the efficiency of using the developed software "Side natural lighting" with manual calculation in accordance with the regulated method [1]. When comparing the amount of time spent on calculating the skylights of the same room using the manual method and the software, the manual method took approximately 55-60 minutes. At the same time, the calculation by the automated method takes 5 minutes and does not require knowledge of the calculation method and formulas.

In addition, a comparison of the accuracy of the automated and manual methods for calculating the side natural lighting of premises was made on the example of a classroom at Khmelnytsky National University. The estimated (ideal) area of the light openings by the manual method is  $S = 14.38 \text{ m}^2$ , and by the automated method -  $S = 18.20 \text{ m}^2$ . Such a large difference is explained by the fact that the "Side lighting" software product does not round off and does not accept approximate values in the calculation process, i.e. it is a more accurate method [2]. The calculation error is then 21%.

It should be borne in mind that each subsequent calculation by the manual method will take approximately the same time as the first (approximately 1 hour), as well as by the automated method (approximately 5 minutes).

The relevance of using an automated method for calculating side natural light is that it completely eliminates the possibility of making an error in calculating formulas, as well as the possibility of calculating side natural light without knowledge of the methodology. In addition, the calculation with the help of such specialized software is 12 times faster and 21% more accurate than the manual method of assessing the side natural light.

Thus, the software product "Side lighting" is gaining its practical importance and application in any sphere of production and non-production sectors of Ukrainian enterprises to check the state of natural side lighting and meet the regulated labor protection standards on this issue, as well as in the construction industry when checking the sufficient state of side natural lighting of residential premises.

## REFERENCES

1. DBN V.2.5:28 - 2018 Natural and artificial lighting - Kyiv: Ministry of Regional Development of Ukraine, 2018. 137 p.
2. Sokolan Iu.S., Shevelya V.V., Piedgon I. Automated approach to assessment of the lateral natural illumination in premises // Ukrainian Journal of Construction and Architecture, No. 5 (011), 2022. - pp. 100-109.

УДК 004.624

## ВИКОРИСТАННЯ СИНТЕТИЧНИХ КОМПОНЕНТІВ ЯК ШАБЛОНІВ ДЛЯ ГЕНЕРУВАННЯ КОДУ

**СТАВИЦЬКИЙ П.В., ВОЙТКО В.В.**

(pavlo.stavytskyi@gmail.com, dekanfki@i.ua )

Вінницький національний технічний університет

*Важливим завданням під час генерування коду є забезпечення декларативного підходу з безпечною типізацією. Таким чином, програма для генерування коду синтаксично є близькою або ідентичною до результуючого згенерованого коду. Однією складовою, що дозволяє вирішити дану проблему під час генерування об'єктно-орієнтованого коду, є використання синтетичних класів, що виконують роль шаблонів з безпечною типізацією для генерування декларувань компонентів об'єктно-орієнтованого програмування.*

**Постановка задачі.** Розглядаючи велику кількість мов програмування загального призначення, можна виділити велику кількість інструментів для метапрограмування та генерування коду. Одним з недоліків багатьох з них є те, що в їх основі лежить підхід імперативного програмування. Зокрема, його результатом є те, що код програми, яка виконує генерування коду, синтаксично значно відрізняється від результуючого згенерованого коду, що збільшує когнітивне навантаження під час роботи з таким програмним забезпеченням та збільшує вірогідність появи помилок [1].

Запропонований метод метапрограмування поєднує декларативність та безпечну типізацію для підвищення швидкості й ефективності генерування програмного коду мов програмування загального призначення. Декларативність полягає в тому, що код програми, яка виконує генерування коду, є близьким або ідентичним до результуючого згенерованого коду. Безпечна типізація в такому контексті гарантує коректність згенерованого коду та можливість знаходження помилок якомога раніше в процесі розробки програмного забезпечення [2].

**Результати дослідження.** Для реалізації декларативного підходу генерування коду зі збереженням безпечної типізації в рамках даного дослідження розроблено плагін компілятора мови програмування Kotlin. Причиною такого рішення є те, що плагін компілятора дозволяє розширити функціонал мови програмування загального призначення в якості нативних компонентів такої мови. Використовуючи анотації, надається їх імплементація у вигляді окремих операторів такої мови програмування, забезпечуючи нативну інтеграцію як складову мови програмування.

Таким чином, використовуючи можливості плагіна компілятора, існує можливість копіювати код, написаний мовою Kotlin, напряму в згенерований код. Таким чином, є гарантія коректності такого коду, забезпечена компілятором перед копіюванням коду.

Під час реалізації декларативного метапрограмування для об'єктно-орієнтованих компонентів постає необхідність генерування великої кількості класів, що мають схожу загальну структуру але різні деталі, такі як ім'я класу або його полів, наявність тих чи інших функцій в залежності від ситуації тощо. Для того, щоб отримати посилання на об'єкти такого типу незалежно від їх конкретної імплементації, використовується підхід синтетичних компонентів. Для цього декларується клас з необхідним набором функцій, полів тощо, позначений анотацією Synthetic. Така анотація надає компілятору необхідну інформацію для коректної компіляції коду. Далі, під час написання функцій, що генерує код, використовується оператор splice, що позначає код, який необхідно скопіювати в первозданному вигляді в згенерований код. Проте, певні оператори, використані всередині, надають додатковий функціонал та метадані під час генерування коду. Наприклад, оператор synthetic дозволяє позначити ім'я та пакет конкретного екземпляра класу, що декларований за попередньо зазначеним синтетичним шаблоном. Весь код, що посилається на такий синтетичний компонент всередині блоку synthetic, робитиме це в контексті конкретного декларованого компонента (рис. 1).

```
@Synthetic
class MyClass

val code = splice {
    synthetic(type<MyClass>("MyClass1", "com.package.name")) {
        val myClassInstance = declareVal("myClass1Instance", MyClass())
        println(myClassInstance)
    }
}

// Записати 'code' у файл
```

Рисунок 1 – Використання синтетичних компонентів для типізованого посилання на згенеровані класи

У результаті виконання такого коду буде згенеровано наступний фрагмент, зображений на рисунку 2.

```
val myClass1Instance = MyClass1()  
println(myClass1Instance)
```

Рисунок 2 – Код, згенерований з допомогою використання синтетичного компонента

Схожим чином синтетичні компоненти використовуються для декларування класів, що слідує обраному шаблону.

**Висновки.** Таким чином, декларативний підхід до реалізації метапрограмування підвищує зручність процесу генерування коду, а в поєднанні з безпекою типізації підвищує його ефективність. Генерування компонентів об'єктно-орієнтованого програмування потребує додаткових інструментів для реалізації в рамках декларативної парадигми. Використання синтетичних компонентів забезпечує реалізацію типізованих посилань на згенеровані класи.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. SMITH, F., GROSSMAN, D., MORRISETT, G., HORNOF, L., & JIM, T. (2003). Compiling for template-based run-time code generation. *Journal of Functional Programming*, 13(3), 677-708. doi:10.1017/S095679680200463X
2. Ставицький, П. В., Войтко, В. В. Метод декларативного метапрограмування на основі розширення синтаксису існуючих мов програмування. // *Вимірювальна та обчислювальна техніка в технологічних процесах*, м. Хмельницький – №3. 2022 – С.5-11. <https://doi.org/10.31891/2219-9365-2022-71-3-1>

УДК 004.42

### ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ BLOCKCHAIN ДЛЯ СТВОРЕННЯ ЗАХИЩЕНИХ ПЛАТФОРМ ОНЛАЙН ГОЛОСУВАННЯ

ФЕДЧУН К. Ю. (Fedchun.Kyrylo2019@vnu.edu.ua)

Волинський національний університет імені Лесі Українки

*У роботі описано можливості застосування технологій Blockchain для забезпечення безпеки, прозорості, надійності та недоступності до втручання в онлайн голосування. Визначено основні вимоги до системи голосування, що буде базуватись на технології Blockchain.*

У теперішній час найрозповсюдженішою моделлю підрахунку голосів є місцеві, централізовані системи голосування. Такі системи можуть стикатися з різними серйозними проблемами на усіх етапах провадження. Найбільш розповсюджені — це проблеми з недостатньою безпекою, присутність високої ймовірності злому чи вторгнення з подальшою можливістю отримати доступ до бази даних голосів з їхньою заміною або розповсюдження приватної інформації. У деяких централізованих системах та системах голосування можуть бути проблеми із прозорістю голосування, особливо якщо такі системи залежні від зовнішнього контролю і важко гарантувати рівність всіх голосів. Наприклад, складно перевірити, які голоси були зараховані, які ні, і чому саме так здійснювався підрахунок голосів. Це може знизити довіру громадян до системи голосування і підірвати її легітимність.