



КАФЕДРА КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

XV-та МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ «FREE AND OPEN SOURCE SOFTWARE»



Дякуємо за підтримку



13-14 лютого 2024 р.
м. Харків

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

КАФЕДРА КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

XV-ої МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ

«FREE AND OPEN SOURCE SOFTWARE»

13-14 лютого 2024 р.

ХАРКІВ 2024

УДК 004
БК 32.973.202

Матеріали XV-ої Міжнародної науково-практичної конференції «Free and Open Source Software», Харків, 13-14 лютого 2024 р. – Харків: Харківський національний економічний університет імені Семена Кузнеця, 2024. – 148 с.

Представлено матеріали пленарних та секційних засідань XV-ої Міжнародної науково-практичної конференції «Free and Open Source Software». Обговорено основні проблеми, науково-технічні досягнення, впровадження і досвід використання сучасних технологій в області безкоштовних програмних продуктів, а також з відкритим вихідним кодом. Висвітлено основні питання безкоштовного прикладного, серверного програмного забезпечення та прикладного програмного забезпечення з відкритим вихідним кодом, безкоштовних сервісів, в тому числі в контексті кібербезпеки, ліцензування та правові аспекти використання безкоштовного програмного забезпечення. Для фахівців науково-дослідних, комерційних організацій, аспірантів та студентів.

Редакційна колегія:
Старкова О.В. – голова, д.т.н.;
Міхєєв І.А. – к.т.н.;

Відповідальний за випуск:
Старкова О.В.

Роботи надруковані з авторських оригіналів, що надані оргкомітету, за авторської редакції.

Електронний варіант матеріалів конференції доступний на сайті конференції:

<https://foss.kn-it.info/>

ФЕНОМЕН АРІ ЯК ФАКТОР СИНЕРГІЇ У МІЖСИСТЕМНІЙ КОМУНІКАЦІЇ ІНТЕЛЕКТУАЛЬНИХ КОМПОНЕНТІВ БІЗНЕС- СИСТЕМ	115
<i>Андрейчіков О.О., Старкова О.В.</i>	
ЗАСТОСУВАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗАЦІЇ ЙМОВІРНІСНОГО ПІДХОДУ ДО ВИЗНАЧЕННЯ ПРАЦЕЗДАТНОСТІ СИСТЕМИ	116
<i>Бугай І.С., Солодовник Г.В.</i>	
ЗАСТОСУВАННЯ ЗАСОБІВ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ПРИЙНЯТТІ РІШЕНЬ В УМОВАХ НЕВИЗНАЧЕНОСТІ	120
<i>Вертебний М.М., Солодовник Г.В.</i>	
ОГЛЯД МОЖЛИВОСТЕЙ ЗАСТОСУНКУ BLENDER ДЛЯ ПРОЦЕДУРНОГО МОДЕЛЮВАННЯ	122
<i>Гречишкін Д.С., Яковлева О.В.</i>	
АВТОМАТИЗАЦІЯ ВИБОРУ ЗАХОДІВ БЕЗПЕКИ В УМОВАХ НЕВИЗНАЧЕНОСТІ	125
<i>Карабан О.Д., Солодовник Г.В.</i>	
АВТОМАТИЗАЦІЯ ПРИЙНЯТТЯ БАГАТОЕТАПНИХ РІШЕНЬ В УМОВАХ РИЗИКУ	127
<i>Карабан О.Д., Солодовник Г.В.</i>	
РОЗРОБКА ПІДСИСТЕМИ ЗАХИСТУ ПЕРСОНАЛЬНИХ ТА ІГРОВИХ ДАНИХ КОРИСТУВАЧА МОБІЛЬНОЇ ГРИ	129
<i>Кошаренко Д.С., Мерлак О.В.</i>	
ОСОБЛИВОСТІ ОЦІНЮВАННЯ МЕТРИК НАТИВНИХ ANDROID ЗАСТОСУНКІВ	130
<i>Макарова Л.М., Татаренко М.А.</i>	
PER8 ТА «CLEAN CODE» В ОЦІНЮВАННІ ЯКОСТІ ЗАСТОСУНКІВ НА PYTHON	132
<i>Макарова Л.М., Штаба В.Г.</i>	
ТЕХНОЛОГІЯ РОЗПІЗНАВАННЯ МОВЛЕННЯ ДЛЯ ПОКРАЩЕННЯ КОРИСТУВАЦЬКОГО ДОСВІДУ В ОСВІТНЬОМУ ДОДАТКУ	134
<i>Пироженко М.Ю., Вишняк М.Ю.</i>	

PEP8 ТА «CLEAN CODE» В ОЦІНЮВАННІ ЯКОСТІ ЗАСТОСУНКІВ НА PYTHON

Макарова Л.М., Штаба В.Г.

E-mail: lidia.makarova@nuos.edu.ua, shtaba09@gmail.com

Миколаїв, Національний університет кораблебудування імені адмірала Макарова

Стандартизація певних процесів написання коду ІТ-проектів відіграє дуже важливу роль в оцінюванні якості коду, з метою забезпечення його розвитку в майбутньому. Стандартизація дозволить більшій кількості поколінь розробників вдосконалювати та розвивати код. Розвиток та удосконалення коду - це одне із найважливіших завдань розробників на довгострокових і масштабних проектах. Загалом для розуміння придатності початкового та напрацьованого коду для проектів, потрібно виконати якісну оцінку цього коду, визначити його зрозумілість та відповідність певним стандартам, та дати оцінку якості і придатності для розвитку, що дозволить в майбутньому більш гнучко та якісно застосовувати бюджет, та виконувати покращення сервісів та продуктів, які використовують програмне забезпечення, що написано на основі цього коду. У цій роботі поставлено завдання визначити фундаментальні стандарти та вимоги для коду мовою Python.

Концепція «Clean Code» - це концепція фундаментального підходу до правил оформлення та написання програмного коду. Ця концепція максимально узагальнена та має певні правила, що підходять до будь-якої високорівневої мови програмування. Дотримання правил «Clean Code» - це ключовий компонент успішної розробки програмного забезпечення. Він покращує продуктивність команди, полегшує процес супроводу і підтримки застосунків, знижує кількість помилок [1, 2]. Тож зрозуміло, що якість коду безпосередньо пов'язана з якістю продукту. Чистий, структурований і зрозумілий код робить застосунок надійнішим, покращує його продуктивність і полегшує внесення змін. У підсумку, він сприяє створенню продукту, який задовольняє потреби користувачів і клієнтів.

Основні засади концепції:

1. Читабельність коду. Код повинен бути написаний так, щоб його було легко читати та розуміти. Це означає використання зрозумілих назв змінних та функцій, а також написання коротких та зрозумілих коментарів.

2. Простота дизайну. Уникнення надмірно складних конструкцій. Часто простіші рішення є надійнішими та легшими для підтримки.

3. DRY (Don't Repeat Yourself). Уникнення дублювання коду. Використання функцій та модулів для уникнення повторення однакових або дуже схожих блоків коду.

4. Тестування. Написання тестів для свого коду, включаючи юніт-тести, щоб забезпечити його надійність та полегшити подальшу розробку та рефакторинг.

5. Рефакторинг. Регулярний перегляд та вдосконалення коду, щоб він залишався чистим та ефективним. Рефакторинг включає в себе виправлення "кодового боргу" (technical debt), оптимізацію та видалення зайвого коду.

6. Відповідність SOLID принципам. Це п'ять принципів об'єктно-орієнтованого програмування, які сприяють створенню більш зрозумілого, гнучкого та підтримуваного коду.

7. Обробка помилок. Написання чіткого та інформативного коду обробки помилок, уникання загальних виключень, де це можливо, і забезпечення того, щоб помилки не залишалися непоміченими.

8. Форматування коду та стандарти. Дотримання усталених стандартів форматування, таких як PEP8 для Python, щоб код був послідовним і легко читався іншими розробниками.

PEP8 - «Style Guide for Python Code» - є документом, який визначає засади стилю написання комп'ютерного коду мовою Python. Цей стандарт був розроблений Гвідо ван Россумом, Баррі Варсоу і Ніком Когланом і вперше опублікований у 2001 році. Він містить рекомендації щодо того, як формувати Python код, щоб він був більш читабельним та уніфікованим [3].

Основа гайду полягає в наступних елементарних правилах:

1. Відступи. Використання 4 пробілів на кожен рівень відступу для візуального розмежування блоків коду.

2. Довжина рядка. Обмеження максимальної довжини рядка 79 символами для коду і 72 для коментарів та докстрінгів (блоків документації). Це забезпечує зручність читання коду на різних пристроях та редакторах.

3. Імпорти. Імпорти мають бути на окремих рядках і групуватися в наступному порядку: стандартна бібліотека, сторонні бібліотеки, локальні модулі програми. Крім того, рекомендується використовувати абсолютні імена для імпорту, де це можливо.

4. Пробіли в виразах та інструкціях. Уникання використання великої кількості пробілів.

5. Коментарі. Коментарі, що не відносяться до коду, повинні бути повністю виправдані. Вони повинні пояснювати контекст або причину для того, що робить код, а не описувати очевидні речі.

6. Іменування конвенції. Використовуйте конвенції іменування для класів, функцій, методів, змінних та інших елементів коду. Наприклад, імена класів повинні бути в CamelCase, а імена функцій та змінних - в lowercase_with_underscores.

7. Строки документації (Docstrings). Використання потрібних лапок для рядків документації. Вони повинні пояснювати, що робить функція, метод або клас, і повинні слідувати специфікації Docstring PEP257.

PEP257 – це стандарт розміщення документації (docstrings) в Python коді, котра розміщується на початку модулів, класів, функцій або методів та дає коротке, але зрозуміле представлення про входні параметри, повернення значень, функції та можливе застосування методу та іншу інформацію. Дотримання такого стандарту дозволить новим програмістам, що тільки нещодавно прийшли на проєкт, краще і швидше розібратися з написаним кодом та вдосконалити його.

Виходячи з розглянутих визначень стандартів та концепцій, можна зробити висновок, що застосування розглянутих особливостей стандартів та їхніх правил дозволить вибудувати та обрати актуальні метрики для мови програмування Python. Це дозволить розробити математичну модель для оцінювання якості коду, придатності його до удосконалення та планування розробки нового функціоналу застосунку, що написаний з його використанням.

Головною метою роботи є проведення практичних досліджень реального програмного коду з застосуванням існуючих математичних моделей та застосунків для збирання метрик. Можливо виявиться необхідним написання власних застосунків та програмних інструментів для збору метрик. Важливим аспектом роботи є розробка математичних моделей або адаптація існуючих моделей задля отримання більш достовірних результатів обробки інформації з метрик Android застосунків та їх відповідність стандартам розробки сучасного програмного забезпечення.

Результати досліджень і розробок математичних моделей можуть бути застосовані для планування етапів та розрахунку часу розробки, визначення кошторису та можливих ризиків на етапах розробки Android застосунків мовою програмування Python.

Література

[1] Про поняття “чистий код” у розробці ПЗ [Електронний ресурс]. – Режим доступу до ресурсу: <https://foxminded.ua/shcho-take-chystyi-kod/>

[2] Mariano Anaya Clean Code in Python: Develop maintainable and efficient code, 2nd Edition 2nd ed. Edition. Packt>, Birmingham - Mumbai, 2021. 422 p. ISBN: 978-1800560215

[3] PEP 8 – Style Guide for Python Code [Електронний ресурс]. – Режим доступу до ресурсу: <https://peps.python.org/pep-0008/>

**Матеріали XV-ої Міжнародної науково-практичної конференції
«FREE AND OPEN SOURCE SOFTWARE»**

Харківський національний економічний університет імені Семена Кузнеця

Відповідальний за випуск: Старкова О.В.

Редактор: Міхєєв І.А.

Затверджено засіданням кафедри кібербезпеки та інформаційних технологій
ХНЕУ імені С. Кузнеця
протокол № 11 від «16» лютого 2024 р.

Видавець і виготовлювач – ХНЕУ імені С. Кузнеця, 61166, м. Харків, просп.
Науки, 9-А
Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру
ДК № 4853 від 20.02.2015 р.