

## ОСОБЛИВОСТІ ПРОЕКТУВАННЯ КОМПОНЕНТІВ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОРГАНІЗАЦІЇ ТЕСТУВАННЯ ЗНАНЬ

Вінницький національний технічний університет

### Анотація

*Розглянуто процес проектування інформаційної технології організації процесу тестування знань. Наведено особливості проектування, наведено UML діаграми інформаційної технології, що описують відповідні програмні модулі.*

**Ключові слова:** проектування інформаційних систем, UML, тестування знань.

### Abstract

The process of designing information technology for the organization of the knowledge testing process is considered. Features of design are given, UML diagrams of information technology describing the corresponding software modules are given.

**Keywords:** design of information systems, UML, testing of knowledge.

### Вступ

Реалії сьогодення потребують постійного навчання, як в професійних сферах розвитку, так і в особистісних, що, в свою чергу, потребує контролю знань. Так як сучасний темп життя досить швидкий та мобільний – онлайн-тестування стає одним із ефективних способів контролю засвоєння набутих знань. Наразі існує багато програмних платформ для організації онлайн тестування учнів та студентів. Проте, недостатньо звертається увага на розробку більш універсальних та гнучких програмних платформ, зокрема для тестування професійних знань, наприклад, за результатами стажування в ІТ-компаніях, тощо. Також, й досі досить складною задачею залишається ефективна організація перевірки відповідей користувачів у відкритих тестах. Саме розробці інформаційної технології організації процесу мультикористувацького тестування знань з орієнтацією на відкриті тести і присвячено дане дослідження [1].

### Проектування інформаційної технології

Вказана розробка створюється для підвищення швидкодії процесів створення тестів, організації мультикористувацького тестування, зокрема, віддалених користувачів, підвищення точності перевірки запитань відкритого типу, шляхом розробки та використання інформаційної технології організації процесу тестування знань.

Проектування – це перевірена та загальноприйнята інженерна техніка. Наприклад, в архітектурі будівель ми розробляємо архітектурні моделі будинків та хмарочосів, щоб допомогти візуалізувати кінцеві продукти. При проектуванні інформаційних продуктів, розділивши рішення на кілька модулів/проектів, коли воно стає занадто вимогливим, перевага полягає в тому, що кожна UML-модель чи діаграма, що описує відповідний програмний модуль, може бути реалізована будь-якою мовою програмування, яка найбільше відповідає визначеним критеріям [2]. Загалом, необхідність створення діаграм класів, наслідування та композиційних зв'язків, є основою для успішної реалізації інформаційної технології.

Діаграми класів використовуються не лише для візуалізації статичного вигляду системи, але й вони також використовуються для побудови виконуваного коду, для прямого та зворотного проектування інформаційної системи [3] . В ході нашої розробки, створимо основний базовий клас User, що буде містити загальну інформацію про користувача. Кожен клас має свій набір атрибутів та методів класу. Для базового класу User атрибутами є: id (унікальний ідентифікатор), first\_name (ім'я), last\_name (прізвище), fathers\_name (по-батькові), age (вік), email (електронна пошта), role (роль). Крім того, базовий клас має перелік методів, деякі з яких, є віртуальними:

- getInfo – отримати інформацію про користувача;
- setParams – встановити значення атрибутів;
- updateInfo – оновити значення атрибутів.

Так як клас User є базовим, від нього наслідуються два інших класи – Examiner і TestUser, які реалізують в собі різницю між ролями користувачів. Під час наслідування додаються нові атрибути та реалізуються віртуальні функції. Таким чином, реалізується класова діаграма наслідування, зображена на рисунку 1.

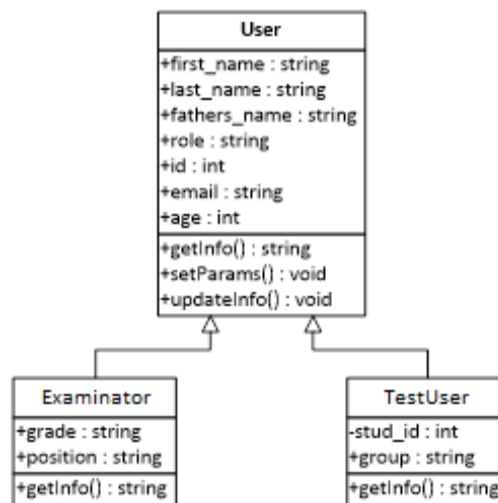


Рисунок 1 – Діаграма наслідування

Крім того, одними із важливих компонентних складових є тести, що реалізуються через клас Test, який має у собі масив запитань. Так як самі запитання, це не примітивні набори рядків, необхідно визначити клас Question, від наявності якого, залежить клас Test. Кожне запитання може в собі містити масив варіантів класу Option, класова діаграма зображена на рисунку 2:

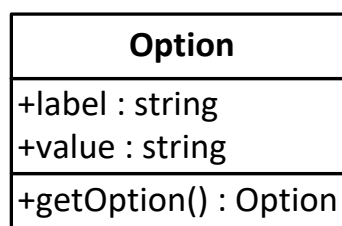


Рисунок 2 – Діаграма класу Option

Змоделюємо відношення двох класів (Test і Question) на діаграмі композиції [4], що зображена на рисунку 3:

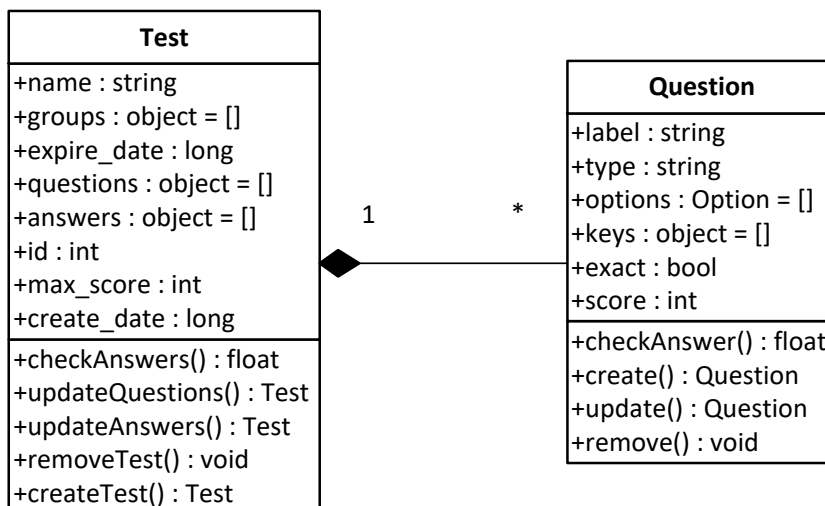


Рисунок 3 – Діаграма композиції

### Висновок

Таким чином, відзначено актуальність створення інформаційної технології організації процесу мультикористувацького тестування знань з орієнтацією на відкриті тести. Розглянуто процес проектування інформаційної технології організації процесу тестування знань. Наведено особливості проектування, UML діаграми інформаційної технології, що описують відповідні програмні модулі. Також відзначено, що проектування та створення діаграм є важливою частиною в розробці інформаційної технології організації процесу тестування знань. Адже це забезпечує можливість спланувати наперед взаємозв'язки між екземплярами класів, їх поведінку та залежності. Крім того, створення діаграм такого виду дає повну картину в особливостях та можливостях використання програмного продукту.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Король, Д., Яровий, А., & Ольшанська, О. 2021 Маг 9. ОСОБЛИВОСТІ СТРУКТУРНОЇ ПОБУДОВИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОРГАНІЗАЦІЇ ТЕСТУВАННЯ ЗНАНЬ. НТКП ВНТУ. Факультет інформаційних технологій та комп'ютерної інженерії. [Електронний ресурс]. - Режим доступу до ресурсу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2021/paper/view/12501/10461>
2. Петрик М.Р. Моделювання програмного забезпечення : науковометодичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с.
3. Методичні матеріали з інформатики. Діаграми класів [Електронний ресурс] – Режим доступу до ресурсу: <https://ua5.org/oop/392-diagrami-klasiv.html>
4. Composite structure diagram [Електронний ресурс] // Visual Paradigm. – 2021. – Режим доступу до ресурсу: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-composite-structure-diagram/>.

**Король Діана Сергіївна** – студентка групи 2КН-20м, факультет інформаційних технологій і комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, Україна, [diana999anaid@gmail.com](mailto:diana999anaid@gmail.com).

**Науковий керівник: Яровий Андрій Анатолійович** – доктор технічних наук, професор, завідувач кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця, Україна, e-mail: [a.yarovyy@vntu.edu.ua](mailto:a.yarovyy@vntu.edu.ua).

**Ольшанська Ольга Вікторівна** – асистент кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця, Україна.

**Korol Diana S.** – Student, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, Ukraine, e-mail: [diana999anaid@gmail.com](mailto:diana999anaid@gmail.com).

**Supervisor: Yarovyi Andrii A.** – Doctor of Sciences (Eng.), Professor, Head of the Department for Computer Science, Vinnytsia National Technical University, Vinnytsia, Ukraine, e-mail: [a.yarovyy@vntu.edu.ua](mailto:a.yarovyy@vntu.edu.ua).

**Olshanska Olga V.** – Assistant, Department for Computer Science, Vinnytsia National Technical University, Vinnytsia, Ukraine.