

ринку праці // Інновації в науці: зб. ст. по матер. XXXVI міжнар. наук.-практ. конф. № 8 (33). - Новосибірськ: СіБАК, 2014.

1. «Сім навичок професійного програміста» Електронний ресурс: <https://habr.com/post/188430/>

2. «Знання и навыки, полезные для каждого программиста» Електронний ресурс: <https://proglib.io/p/skills-for-programmers/>

3. Work.ua Електронний ресурс: <https://www.work.ua>

4. ТОП-10 качеств программиста. Мнение лучших работодателей Електронний ресурс: [https://geekbrains.ru/posts/programmer\\_top\\_qualities](https://geekbrains.ru/posts/programmer_top_qualities)

## МОЖЛИВОСТІ МОВИ UML ДЛЯ ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

*Єремєєв Володимир Сергійович,*

*доктор технічних наук, професор кафедри інформатики і кібернетики*

*Сердюк Ірина Миколаївна,*

*асистент кафедри інформатики і кібернетики*

*Курлянський Сергій Сергійович,*

*студент магістратури.*

*Мелітопольський державний педагогічний університет*

*імені Богдана Хмельницького*

**Анотація.** Останнім часом для моделювання різноманітних процесів, у тому числі для опису програмного забезпечення на різних рівнях використовується уніфікована мова моделювання (UML). У статті здійснюється огляд особливостей уніфікованої мови моделювання, наводиться структура діаграми класів як однієї з поширених діаграм мови UML.

**Ключові слова:** мова UML, програмне забезпечення, проектування, система, CASE-засоби, діаграма класів

Процес проектування програмного забезпечення можна представити у вигляді створення різноманітних моделей, які відображають структуру, функціональні можливості, взаємодію елементів системи із зовнішнім середовищем тощо. Останнім часом для моделювання різноманітних процесів, у тому числі для опису

програмного забезпечення на різних рівнях використовується уніфікована мова моделювання (UML).

Метою статті є загальний огляд можливостей мови UML для проектування програмного забезпечення.

Під проектуванням програмного забезпечення розуміється процес створення специфікацій програмного забезпечення на основі вимог, які пред'являються до нього перед початком проектування. В основі проектування програмного забезпечення є системний підхід, під яким мається на увазі методологія дослідження об'єкта будь-якої природи як системи. Для того, щоб забезпечити представлення системи, зокрема програмного забезпечення, з різних боків, використовується різноманітні моделі. При цьому процес створення моделі називається моделюванням.

Моделі використовуються з метою осмислення структури та поведінки майбутньої системи, зокрема програмного забезпечення. Отримані у процесі опису і вивчення моделі дані надають змогу розробникам зменшити можливі ризики щодо користування програмним продуктом, підвищити його функціональні можливості, забезпечити взаємодію із користувачами, створити супровідну документацію тощо. При розробці програмного забезпечення використовуються різні моделі, найбільш наочна з яких – графічна. В свою чергу, існує багато видів графічних моделей (блок схеми, діаграми, фреймові моделі, семантичні мережі тощо). Вони дозволяють описати проблеми за допомогою зрозумілих абстракцій, що відтворюють елементи програмного забезпечення, об'єкти та події реального світу.

Для успішного проектування програмного забезпечення необхідно правильно визначитися із мовою модулювання. Однією із поширених мов є уніфікована мова моделювання (UML), під якою розуміється уніфікована загальноцільова мова візуального моделювання, яка призначена для візуалізації, проектування та документування бізнес-процесів, різноманітного програмного забезпечення [2, с. 209], у тому числі складних інформаційних систем та програмних комплексів.

Версія UML 1.4.2 прийнята в якості міжнародного стандарту ISO/IEC 19501:2005. В кінці 2004 р була представлена друга версія UML. Домінуючою характеристикою цієї версії стала підвищена

точність визначення мови, що дозволяє використовувати більш високий рівень автоматизації [1, с. 115]. UML 2.4.1 прийнятий як міжнародний стандарт ISO / IEC 19505-1, 19505-2. Остання версія 2.5. була розроблена у червні 2015 року [5].

Важливою складовою мови UML є графічна нотація, яку можна визначити як своєрідний синтаксис мови, який дозволяє проектувати програмне забезпечення через візуалізацію його об'єктів, їх поведінки та функціональних можливостей. Нотація – сукупність символів і правил їх застосування, що використовуються для представлення понять та зв'язків між ними [4, с. 279]. Крім того, слово «нотація» підкреслює, що UML – мова графічна, де моделі малюють на папері або за допомогою спеціальних CASE-засобів, а не записують у вигляді рядків, формул тощо. Серед CASE засобів, що дозволяють створювати UML-діаграми, найбільш відомими є IBM Rational Rose, Microsoft Visio, Sparx Systems Enterprise Architect та інші.

У термінах мови проектувальник створює діаграми (diagram), які можна представити у вигляді спеціальних графічних конструкцій, які описують різні стани системи. Для створення моделі в UML використовується чотири види елементів нотації: фігури, лінії, значки, написи. Зазначимо, що мова UML дозволяє генерувати програмний код для середовищ програмування з використанням Case-засобів, проте власне сама мовою програмування не вважається.

При проектуванні об'єктно-орієнтованих систем діаграми класів є обов'язковими. Вона складається з набору статичних елементів моделі, які її декларують. При побудові діаграми класу можна використовувати такі об'єкти як клас (іменованний опис сукупності об'єктів, що володіють однаковою структурою, семантикою, властивостями та відносинами з іншими об'єктами. На діаграмі зображується у вигляді прямокутника), атрибут (іменована властивість класу), операція (іменована абстракція того, що можна робити з об'єктом), зв'язок між класами. Дано визначення та коротко охарактеризуємо вищезазначені поняття [3, с. 13].

У діаграмі класів можуть брати участь зв'язки трьох різних категорій: dependency (залежність), generalization (узагальнення) та association (асоціація). Зупинимося лише на третьому типі зв'язку. Асоціацією називається структурний зв'язок, який показує, що об'єкти

одного класу певним чином пов'язані з об'єктами іншого або того ж самого класу.

Слід додати, що в діаграмі класів можна відобразити ту ситуація, коли асоціація між двома класами має спеціальний вид «частина-ціле». Така асоціація називається агрегатною, яка графічно зображується у вигляді простої асоціації з незафарбованим ромбом на стороні класу-цілого. З іншого боку, існують випадки, коли знищення елемента-«цілого» призводить до знищення всіх його елементів-«частин». В такому випадку агрегатні асоціації з такими властивостями називаються композитними. Графічно композиція представляється у вигляді простої асоціації, доповненої зафарбованим ромбом, який розташований з боку елемента-«цілого».

Отже, уніфікована мова моделювання за рахунок різноманітних діаграм, які є графічними нотаціями мови, здатна повністю задовольнити розробника програмного забезпечення на етапі проектування. Мова продовжує розвиватися, про що свідчить поява нових версій. Діаграма класів, яка є однією з частин повної моделі інформаційної системи, дозволяє спроектувати об'єкти наочної області, зв'язки між ними, характеристики та поведінку.

#### *Література*

1. Брауде Э. Технология разработки программного обеспечения. СПб.: Питер, 2004. 655 с.
2. Станкевич І. В., Тігарєва В. А. Переваги та недоліки застосування уніфікованої мови uml під час моделювання бізнес-процесів складних організацій // Наука й економіка. 2014. № 4. С. 207-216.
3. Шаров С. В., Шамардак О. А. Концептуальне моделювання інформаційної системи за допомогою мови UML // Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення. 2014. №7. С. 10-14.
4. Томашевський О.М., Цегелик Г.Г., Вітер М.Б., Дудук В.І. Інформаційні технології та моделювання бізнес-процесів: навч. посіб. К.: «Видавництво «Центр учбової літератури», 2012. 296 с.
5. UML: сайт. URL: <http://www.wikiwand.com/ru/UML> (дата звернення: 24.04.2018).