

КУНДОС МАКСИМ

Приватний вищий навчальний заклад
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'ячука»
<https://orcid.org/0009-0001-0310-357X>
e-mail: kundosm@gmail.com

СОЛОВЕЙ ЛЮДМИЛА

Приватний вищий навчальний заклад
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'ячука»
<https://orcid.org/0009-0001-2832-1741>
e-mail: lyuda_solovej@ukr.net

ГРИСЮК АНДРІЙ

Приватний вищий навчальний заклад
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'ячука»
<https://orcid.org/0009-0007-2287-5982>
e-mail: jdroidcoder@gmail.com

ЗАСТОСУВАННЯ СУЧАСНИХ МЕТОДІВ І ТЕХНОЛОГІЙ МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ПІДГОТОВЦІ ІТ-ФАХІВЦІВ

У статті розглянуті методологічні підходи до моделювання програмного забезпечення. Запропоновані методи та інструменти моделювання з розробки програмного забезпечення для ІТ проєктів. Визначено важливість розуміння основ моделювання та аналітики для сучасних ІТ-фахівців. В основі дослідження лежить аналіз сучасних методів і технологій, таких як Model-Driven Development (MDD) і Unified Modeling Language (UML), які сприяють ефективному створенню та управлінню складними програмними системами.

Основна увагу зосереджено на важливості візуалізації вимог, оптимізації проєктування, а також покращення комунікації між командними розробниками. Зокрема, підкреслюється, що моделювання допомагає виявляти помилки на ранніх етапах, що значно знижує ризики і витрати, пов'язані з їх виправленням у пізніших стадіях розробки програмного забезпечення.

Дослідження аналізує вплив адаптивних методологій, таких як Agile і DevOps, на моделювання, відзначаючи його роль у забезпеченні гнучкості та швидкої реакції на зміни вимог. Результати дослідження підтверджують, що володіння сучасними підходами до моделювання є необхідною складовою професійної підготовки майбутніх ІТ-фахівців, що дозволяє їм успішно адаптуватися до динамічного ІТ-ринку. Моделювання є одним із ключових етапів процесу розробки програмного забезпечення, що забезпечує формалізацію вимог, структурування компонентів системи та створення основи для подальшого проєктування та реалізації. Сучасні методи та технології моделювання надають змогу не лише підвищити ефективність процесу розробки, але й забезпечити гнучкість систем у відповідь на динамічні зміни бізнес-вимог. Використання сучасної методології моделювання є важливим етапом в освітньому процесі, адже допомагає здобувачам освіти не лише розуміти основи проєктування, але й готувати їх до викликів реальних проєктів у сфері інформаційних технологій.

Ключові слова: програмне забезпечення, моделювання програмного забезпечення, Model-Driven Development (MDD), Unified Modeling Language (UML), гнучкі методології, адаптивні моделі, якість програмного забезпечення, DevOps, Agile.

KUNDOS MAKSYM, SOLOVEI LIUDMYLA, HRYSYUK ANDRIY

The private higher educational institution
«Academician Stepan Demianchuk International
University of Economics and Humanities», Rivne

APPLICATION OF MODERN METHODS AND TECHNOLOGIES OF SOFTWARE MODELING IN THE TRAINING OF IT SPECIALISTS

Methodological approaches to software modeling are considered in the article. Proposed modeling methods and tools for software development for IT projects. The importance of understanding the basics of modeling and analytics for modern IT specialists is determined. The research is based on the analysis of modern methods and technologies, such as Model-Driven Development (MDD) and Unified Modeling Language (UML), which contribute to the effective creation and management of complex software systems.

The focus is on the importance of visualizing requirements, optimizing design, and improving communication between team developers. In particular, it is emphasized that modeling helps to detect errors in the early stages, which significantly reduces the risks and costs associated with their correction in the later stages of software development.

The study analyzes the impact of adaptive methodologies such as Agile and DevOps on modeling, noting its role in enabling flexibility and rapid response to changing requirements. The results of the study confirm that the mastery of modern approaches to modeling is a necessary component of the professional training of future IT specialists, which allows them to successfully adapt to the dynamic IT market. Modeling is one of the key stages of the software development process, which ensures the formalization of requirements, the structuring of system components, and the creation of a basis for further design and implementation. Modern modeling methods and technologies make it possible not only to increase the efficiency of the development process but also to ensure the flexibility of systems in response to dynamic changes in business requirements. The use of modern modeling methodology is an important stage in the educational process because it helps students not only understand the basics of design but also prepares them for the challenges of real projects in the field of information technologies.

Keywords: software, software modeling, Model-Driven Development (MDD), Unified Modeling Language (UML), flexible methodologies, adaptive models, software quality, DevOps, Agile.

Постановка проблеми

Сучасна ІТ-галузь постійно розвивається, що вимагає від ІТ-фахівців не лише глибоких теоретичних знань, а й практичних навичок у моделюванні, розробці та тестуванні програмного забезпечення. У зв'язку з цим, важливою частиною підготовки ІТ-фахівців є навчання на основі сучасних методів і технологій моделювання програмного забезпечення. Ці методи дозволяють засвоювати принципи проектування, тестування та впровадження програмних систем, застосовуючи їх у реальних сценаріях, що значно підвищує їхню кваліфікацію на ринку праці.

Моделювання програмного забезпечення вирішує кілька ключових проблем, які виникають під час розробки складних інформаційних систем. Однією з найпоширеніших проблем, з якими стикаються розробники, є неясність вимог. Часто замовники не можуть чітко сформулювати, що саме їм потрібно, в результаті, це призводить до непорозуміння і незадоволення кінцевим продуктом, надлишкових затрат ресурсів та недотримання запланованих термінів виконання.

Аналіз останніх досліджень.

Дослідження питань моделювання програмного забезпечення проводили багато вчених та фахівців у галузі комп'ютерних наук та інформаційних технологій, які внесли значний вклад у розвиток теорії та практики моделювання програмного забезпечення, що сприяло формуванню сучасних методів розробки.

У [1] досліджено, як принципи теорії обмеження, запропоновані Eliyahu M. Goldratt, можуть допомогти оптимізувати процеси розробки програмного забезпечення, особливо шляхом ідентифікації та усунення «вузьких місць» (bottlenecks).

Використовуючи методи, такі як Model-Driven Development (MDD), розробники можуть значно полегшити проектування, автоматизуючи частину процесу і знижуючи ймовірність людських помилок

В роботі [2] проведено аналіз та визначено перспективи впровадження розробленої моделі системи управління якістю процесу розробки програмного забезпечення на основі процесного підходу PDCA (Plan-Do-Check-Act) українськими ІТ-компаніями, що займаються розробкою програмного забезпечення.

Дослідження [3] зосереджене на проблематиці управління ризиками в процесі розробки програмного забезпечення, проаналізовано сучасні підходи до оцінки ризиків, особливо в контексті суб'єктивних і об'єктивних чинників, які впливають на результати розробки проєктів.

У статті [4] запропоноване спеціалізоване програмне забезпечення моделювання, яке дозволяє дослідити різні режими управління динамічною консолідацією віртуальних машин, забезпечує протоколювання результатуючої інформації, такої, як показники продуктивності та діаграми навантажень, а також дозволяє визначати оптимальні параметри моделі для різних режимів роботи центрів обробки даних, мінімізуючи кількість активних фізичних серверів та зменшуючи кількість порушень SLA.

Сидорова М.Г., Байбуз О.Г. і Лапєць О.В. розглядають важливість розвитку комунікативних навичок під час розробки програмного забезпечення, також підкреслюють, що успіх ІТ-проєктів залежить не лише від технічних знань, але й від "м'яких" навичок (soft skills), особливо в умовах використання сучасних методологій розробки ПЗ [5].

Формулювання цілей статті

Метою роботи є: обґрунтування доцільності застосування сучасних методів і технологій моделювання програмного забезпечення для підготовки фахівців здатних адекватно реагувати на виклики та сучасні тенденції ІТ галузі.

Виклад основного матеріалу.

Для сучасного ІТ-фахівця також важливим стає вміння інтегрувати моделювання в гнучкі методології розробки, такі як Agile та DevOps, що дозволяє швидко адаптуватися до змін у вимогах до ІТ-проєктів та забезпечувати високу якість програмного забезпечення. Застосування цих методологій готує майбутніх ІТ-фахівців до роботи в умовах динамічного ринку, де ефективність розробки й адаптивність до змін є визначальними факторами для успішної кар'єри в індустрії інформаційних технологій.

Моделювання, зокрема за допомогою UML (Unified Modeling Language), дозволяє створювати візуальні діаграми, які спрощують спілкування між технічними та нетехнічними учасниками проєкту, забезпечуючи більш точне розуміння вимог замовника. Завдяки цьому можна спрощувати складні концепції, роблячи їх зрозумілими як для технічних, так і для нетехнічних учасників проєкту. Наприклад, діаграми випадків використання дають змогу замовнику побачити, які саме функціональні можливості забезпечуватиме система, тоді як діаграми класів допомагають розробникам уявити структуру об'єктів, що будуть використовуватися в коді. Створення моделей дозволяє тестувати концепції та перевіряти різні сценарії до початку реалізації програмної та інфраструктурної частин, що знижує ризики та витрати, пов'язані з виправленням помилок на пізніших етапах проєкту. Використання діаграм активностей (Activity Diagram) для опису логіки поведінки елементів системи та/або користувачів, дає можливість чітко описати всі процеси, виявити нечітку логіку, а також оперативно вносити зміни у функціональні та нефункціональні вимоги, алгоритми роботи, архітектуру програмного забезпечення. Окрім того, моделі допомагають структурувати проєкт і виявляти «вузькі місця» в процесах. Це особливо важливо для великих систем, де складність зростає

експоненційно, а виявлення помилок на ранніх стадіях дозволяє уникнути переробок і значно знизити вартість створення таких систем.

Ще одна важлива перевага UML полягає в тому, що він забезпечує стандартизацію, що знижує ризик непорозуміння між учасниками проекту та сприяє повторному використанню компонентів. Моделювання з UML дозволяє легше адаптувати проект під зміни, що особливо важливо в умовах динамічного розвитку ринку. Крім того, використання UML для моделювання програмного забезпечення дає можливість автоматично генерувати код та документацію на основі створених діаграм. Це дозволяє скоротити час розробки та мінімізувати кількість помилок, оскільки початковий код генерується на основі вже детально опрацьованих моделей, що значно прискорює розробку та зменшує кількість помилок, полегшує подальшу підтримку, що, в свою чергу, є важливим фактором в умовах постійних змін та конкурентного середовища. Генерація коду також забезпечує кращу синхронізацію між проектною документацією та реалізацією, зменшуючи ймовірність невідповідностей між спроектованим рішенням, і реально розробленим продуктом. Завдяки цьому підходу розробники можуть зосередитися на вдосконаленні логіки системи, використовуючи вже існуючу модель як основу для подальшої роботи. Укладачі посібника [6] С. Ю. Манаков, О. Г. Трофименко, Ю. Г. Лобода та А. І. Дика висвітлюють основні концепції та методи моделювання в розробці програмного забезпечення, охоплюючи теоретичні основи та практичні аспекти, а також інструменти, що використовуються для створення моделей програмного забезпечення.

Сучасні підходи до розробки, такі як Agile, потребують гнучкості в управлінні змінами. Моделі можуть бути швидко адаптовані до нових вимог, що дозволяє командам розробників ефективно реагувати на зміни без значних витрат на передпроекування. Це є особливо важливим у динамічному середовищі, де вимоги можуть змінюватися в процесі реалізації проекту.

Адаптивна методологія DevOps у моделюванні програмного забезпечення дозволяє автоматизувати, прискорювати та спрощувати всі етапи життєвого циклу програмного продукту. Вона сприяє інтеграції та оптимізації процесів, забезпечуючи швидке впровадження змін і безперервне вдосконалення якості продукту.

Таким чином, ці методології дозволяють покращити ефективність розробки програмного забезпечення, прискорити вихід продуктів на ринок, а також підвищити якість та задоволення клієнтів. Разом ці підходи можуть значно підвищити ефективність та швидкість створення і впровадження сучасних програмних продуктів.

Висновки

Застосування сучасних методологічних підходів до моделювання програмного забезпечення дає можливість майбутнім фахівцям в ІТ-галузі працювати у гнучких командах, адаптуватися до змін, використовувати автоматизовані інструменти для управління процесами розробки та операційної підтримки. Розглянуті підходи дозволяють створювати практично орієнтовані курси, які готують студентів до викликів реальної роботи в ІТ-компаніях, де швидкість та ефективність є ключовими факторами успіху.

Література

1. Ribeiro S., Schmitz E., Alencar A., Silva M. Literature Review on the Theory of Constraints Applied in the Software Development Process. IEEE Latin America Transactions. 2018. vol. 16, no. 11, pp. 2747-2756, <https://doi.org/10.1109/TLA.2018.8795116>
2. Кузь М. В, Пікуляк М. В. Остафійчук Т. Д. Модель системи управління якістю процесу розробки програмного забезпечення. Proceedings of the 2019 Scientific Seminar on Innovative Solutions in Software Engineering. Івано-Франківськ, 2019. С. 19-21, <https://doi.org/10.5281/zenodo.4091480>
3. Коваленко О.В. Методи якісного аналізу та кількісної оцінки ризиків розробки програмного забезпечення. Системи управління, навігації та зв'язку. Полтава, 2018. Т. 3(49). С. 116-125. <https://doi.org/10.26906/SUNZ.2018.3.116>
4. Спеціалізоване програмне забезпечення для моделювання динамічної консолідації віртуальних машин / Е.В. Жаріков, С.Ф. Теленик // Проблеми програмування. — 2022. — № 1. — С. 3-12. — Бібліогр.: назв. — укр.
5. Сидорова М. Г., Байбуз О. Г., Лапець О. В. (2021). Аналіз необхідних комунікативних навичок при розробці програмного забезпечення. Актуальні проблеми автоматизації та інформаційних технологій, 25, 152-157. <https://doi.org/10.15421/432115>.
6. Моделювання програмного забезпечення : навч.-метод. посіб. [Електронний ресурс] / уклад.: С. Ю. Манаков, О. Г. Трофименко, Ю. Г. Лобода, А. І. Дика ; Нац. ун-т «Одеська юрид. академія». – Одеса : Фенікс, 2023. – 145 с. Режим доступу: <https://doi.org/10.32837/11300.25952>

References

1. Ribeiro S., Schmitz E., Alencar A., Silva M. Literature Review on the Theory of Constraints Applied in the Software Development Process. IEEE Latin America Transactions. 2018. vol. 16, no. 11, pp. 2747-2756, <https://doi.org/10.1109/TLA.2018.8795116>
2. Kuz M. V, Pikuliak M. V. Ostafichuk T. D. Model systemy upravlinnia yakistiu protsesu rozrobky prohramnoho zabezpechennia. Proceedings of the 2019 Scientific Seminar on Innovative Solutions in Software Engineering. Ivano-Frankivsk, 2019. S. 19-21, <https://doi.org/10.5281/zenodo.4091480>

3. Kovalenko O.V. Metody yakisnoho analizu ta kilkisnoi otsinky ryzykiv rozrobky prohramnoho zabezpechennia. Systemy upravlinnia, navihatsii ta zviazku. Poltava, 2018. T. 3(49). S. 116-125. <https://doi.org/10.26906/SUNZ.2018.3.116>
4. Spetsializovane prohramne zabezpechennia dlia modeliuvannia dynamichnoi konsolidatsii virtualnykh mashyn / E.V. Zharikov, S.F. Telenyk // Problemy prohramuvannia. — 2022. — № 1. — S. 3-12. — Bibliohr.: nazv. — ukr.
5. Sydorova M. H., Baibuz O. H., Lapets O. V. (2021). Analiz neobkhidnykh komunikatyvnykh navychok pry rozrobsi prohramnoho zabezpechennia. Aktualni problemy avtomatyzatsii ta informatsiinykh tekhnolohii, 25, 152-157. <https://doi.org/10.15421/432115>.
6. Modeliuvannia prohramnoho zabezpechennia : navch.-metod. posib. [Elektronnyi resurs] / uklad.: S. Yu. Manakov, O. H. Trofymenko, Yu. H. Loboda, A. I. Dyka ; Nats. un-t «Odeska yuryd. akademiia». — Odesa : Feniks, 2023. — 145 s. Rezhym dostupu: <https://doi.org/10.32837/11300.25952>