

УДК 004.5+004.942

ФОРМАЛІЗАЦІЯ ПРОЦЕСІВ ПРОЄКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ ЗАСОБАМИ UML

А. В. Кудряшова, Ю. О. Білецький

Національний університет «Львівська політехніка»,
вул. Степана Бандери, 12, Львів, 79013, Україна

Здійснено формалізоване представлення структури та функціоналу інформаційної системи моніторингу розвитку дитини за допомогою UML. Розглянуто чотири типи діаграм — варіантів використання, класів, діяльності та послідовності. Побудована діаграма варіантів використання відображає логіку дій користувача — одного з батьків, який взаємодіє з мобільним додатком за п'ятьма основними сценаріями: моніторинг фізичної активності, ведення харчового щоденника, фіксація спогадів, отримання прогнозу та перегляд історії. Кожен сценарій виконується автономно, без звернення до зовнішніх систем. Діаграма класів деталізує структуру проєкту. Виділено класи даних, DAO-компоненти, ViewModel-рівень та інтерфейс. Інтеграція модуля TensorFlow Lite реалізована через окремий клас, що отримує вхідні дані з PredictionViewModel. Типи зв'язків між класами визначено відповідно до логіки взаємодії. Діаграма діяльності фіксує внутрішню послідовність опрацювання запитів користувача. Ініціалізація бази даних і датчиків відбувається одразу після запуску додатка. У подальшому користувач вводить дані щодо харчування, створює текстові та графічні записи, активує модуль персоналізованих рецептів і переглядає збережену історію. Діаграма послідовності демонструє динамічну взаємодію між компонентами. Відображено передачу запитів між користувачем, інтерфейсом, ViewModel і модулями даних. Особливу увагу приділено етапу прогнозування. Сукупність побудованих моделей дозволяє сформулювати чітке уявлення про логіку та структуру системи, забезпечує логічну узгодженість між компонентами, спрощує реалізацію й тестування програмного забезпечення.

Ключові слова: UML, діаграма варіантів використання, діаграма класів, діаграма послідовності, діаграма діяльності, моделювання, інформаційна система, проєктування мобільного додатку.

Постановка проблеми. У сучасних умовах зростає потреба у побудові систем, здатних забезпечити ефективну взаємодію з користувачем та працювати автономно. Для досягнення цих цілей важливо ще на ранніх етапах проєктування мати чітке уявлення про логіку роботи системи, структуру компонентів та способи обміну даними між ними.

Недостатня формалізація таких процесів часто призводить до неузгодженості між рівнями логіки, ускладнює підтримку програмного коду та підвищує ймовірність

помилки. Застосування засобів моделювання UML дозволяє уникнути зазначених проблем, надаючи можливість фіксувати функціональні сценарії, визначати структуру класів, описувати динаміку процесів та зв'язки між модулями. У зв'язку з цим актуальним є дослідження підходів до формалізації процесів проєктування систем за допомогою UML-діаграм.

Аналіз останніх досліджень та публікацій. У сучасних дослідженнях UML є базовим інструментом для моделювання структури та логіки інформаційних систем. У [1] встановлено домінування діаграм класів і варіантів використання в інженерії вимог і архітектурному моделюванні. У [2] проаналізовано підходи до автоматичної генерації UML-моделей на основі користувацьких історій. Виокремлено діаграми класів та діаграми послідовності як основу для проєктування динаміки поведінки систем. У прикладних розробках [3, 4] UML активно використовується для проєктування автономних мобільних додатків. Наприклад, у [3] запроєктовано систему моніторингу місця знаходження дитини із застосуванням UML. У [4] UML використано для проєктування вебзастосунку для ведення щоденника розвитку дитини.

Незважаючи на наявність публікацій щодо формального опису систем за допомогою UML, більшість з них обмежується окремими задачами (структура, взаємодія, інтерфейс). Водночас недостатньо уваги приділено використанню UML-діаграм як базового засобу формалізації всіх рівнів проєктування — від функціональних сценаріїв до передачі даних між компонентами.

Мета статті. Метою дослідження є аналіз особливостей проєктування інформаційних систем засобами UML.

Виклад основного матеріалу дослідження. У процесі створення інформаційних систем проєктування виконує важливу функцію — забезпечує цілісне уявлення про структуру, логіку роботи та взаємодію компонентів. Особливо це важливо в системах, орієнтованих на взаємодію з користувачем, де на етапі проєктування необхідно передбачити повний цикл опрацювання даних. Особливості використання UML (Unified Modeling Language) розглянуто на прикладі проєктування інформаційної системи моніторингу розвитку дитини.

UML дозволяє створювати наочні моделі, що охоплюють функціональну, поведінкову й структурну складові програмного середовища. Серед найпоширеніших — діаграма варіантів використання (Use Case Diagram), діаграма діяльності (Activity Diagram), діаграма послідовності (Sequence Diagram) та діаграма класів (Class Diagram) [5].

На діаграмі варіантів використання (рис. 1) відображаються основні сценарії взаємодії користувача з додатком. Головний актор — в даному випадку один із батьків, що здійснює моніторинг розвитку дитини. Виділено п'ять функціональних сценаріїв, які відповідають базовим потребам: відстеження фізичної активності, введення харчових даних, ведення щоденника, формування зведеної інформації та отримання прогнозів. На UML-діаграмі варіанти використання зображено у вигляді овальних елементів, що пов'язані асоціативними зв'язками із актором «Батьки». Всі функції охоплені прямокутником, що позначає межі системи й підкреслює її автономність. Взаємозалежності між сценаріями, зокрема включення однієї функції в іншу, позначено стрілками типу include.



Рис. 1. Діаграма варіантів використання системи

Діаграма класів (рис. 2) ілюструє статичну структуру інформаційної системи. На ній зображуються основні класи, їх атрибути, методи та типи зв'язків. Архітектурною основою є модель MVVM (Model–View–ViewModel) з поділом логіки на три рівні — дані, інтерфейс та проміжна логіка взаємодії. Класи даних (зокрема ActivityEntry, NutritionEntry) пов'язані з об'єктами доступу до даних (ActivityDao, NutritionDao) за допомогою анотацій Room. DAO (Data Access Object) передають інформацію до ViewModel, звідки дані оновлюють відповідні інтерфейсні фрагменти. Модель машинного навчання на основі TensorFlow Lite реалізована як окремий модуль, що отримує входні параметри з PredictionViewModel. Асоціації між класами позначені стрілками, залежності — пунктирними лініями.

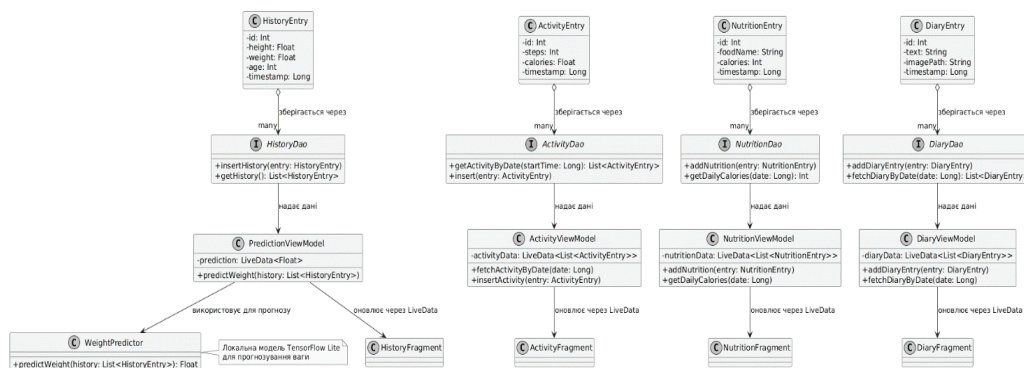


Рис. 2. Діаграма класів

Діаграма діяльності (рис. 3) фіксує послідовність дій в системі під час взаємодії з користувачем.

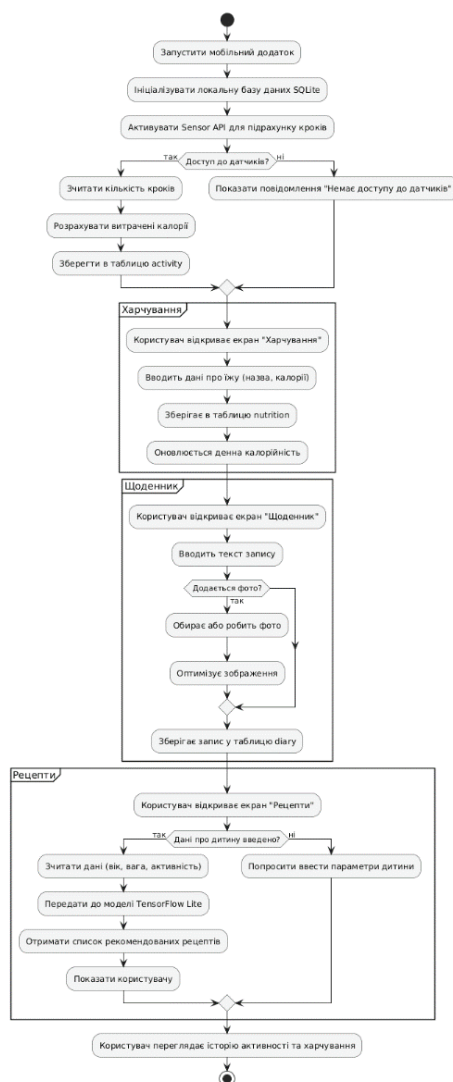


Рис. 3. Діаграма діяльності

Після запуску відбувається ініціалізація локальної бази даних SQLite та активація Sensor API, який відповідає за облік фізичної активності. Користувач також може перейти до модуля харчування для введення даних про спожиті продукти. Система зберігає ці записи та обчислює добову калорійність. У щоденнику доступна можливість створення текстового запису з додаванням зображень. Модуль рецептів активується лише після заповнення ключових параметрів дитини, таких як вік, вага, рівень активності та наявність алергій. Надалі ці значення передаються для формування індивідуалізованого переліку рецептів до моделі на базі TensorFlow Lite. Завершальним етапом є перегляд історії активності та харчування на основі локально збережених даних.

Для демонстрації порядку інформаційного обміну між компонентами системи будується діаграма послідовності (рис. 4). Вона відображає динамічну поведінку системи в основних сценаріях.

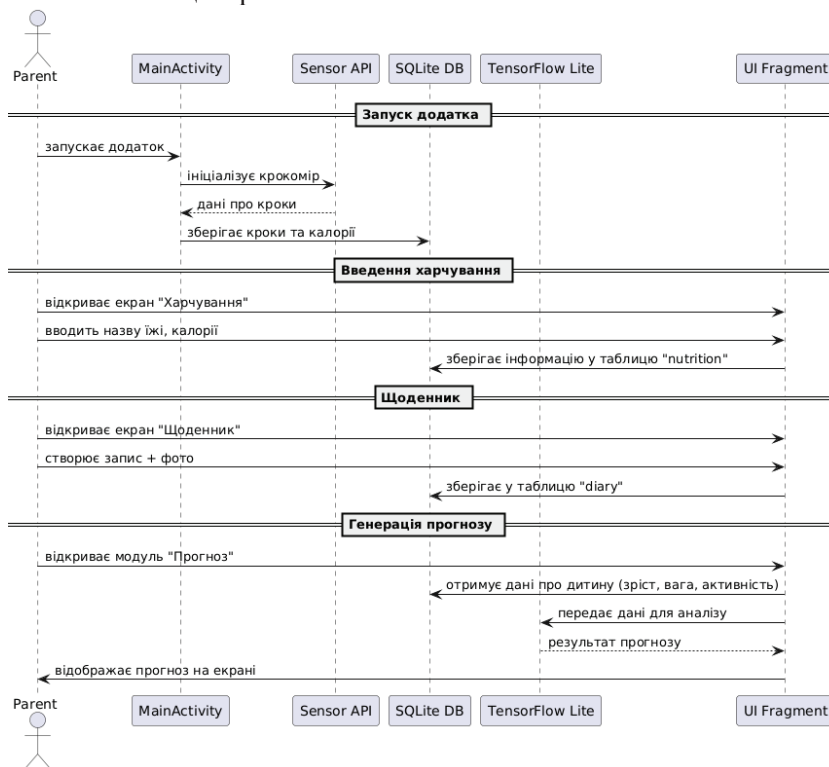


Рис. 4. Діаграма послідовності

Таким чином, сукупність UML-діаграм — варіантів використання, класів, активності та послідовності — формує цілісне формалізоване представлення логіки функціонування й структури проєктованої системи.

Висновки. Здійснено аналіз особливостей проєктування інформаційних систем засобами UML. Розроблено сукупність діаграм, що відображають особливості структури та функціонування додатка: діаграму варіантів використання, яка фіксує взаємодію користувача з системою; діаграму класів, що демонструє архітектуру згідно з MVVM; діаграму діяльності, яка описує послідовність оброблення даних; діаграму послідовності, що ілюструє порядок інформаційного обміну між компонентами. Побудовані моделі забезпечують чіткість у розподілі логіки, структурованість коду та слугують основою для подальшої програмної реалізації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Genero M., Fernández-Saez A. M., Nelson H. J., Poels G., Piattini M. Research review: a systematic literature review on the quality of UML models. *Journal of Database Management*. 2011. № 22 (3). Pp. 46–70.

2. Mornie M. N., Jali N., Junaini S. N., Mit E., Shiang C. W., Saeed S. Visualisation of user stories in UML models: A systematic literature review. *Acta Inform. Pragensia*. 2023. № 12 (2). Pp. 419–438.
3. Indrayana I. N. E., Sutawinaya P., Pratiwi N. M. W. D., Prihatini P. M., Asri S. A. Android-based child monitoring application using a smartwatch and geofence service. *Journal of Physics: Conference Series*. 2021. № 1803 (1), 012024.
4. Sudarmilah E., Saputra D. B., Arbain A. F. B., Murtiyasa B. Web-based system for growth and development monitoring early childhood. *Journal of Physics: Conference Series*. 2021. № 1874 (1), 012024.
5. Шаховська Н. Б., Литвин В. В. Проектування інформаційних систем : навчальний посібник. Львів : «Магнолія-2006», 2011. 380 с.

REFERENCES

1. Genero M., Fernández-Saez A. M., Nelson H. J., Poels G., Piattini M. Research review: a systematic literature review on the quality of UML models. *Journal of Database Management*. 2011. № 22 (3). Pp. 46–70.
2. Mornie M. N., Jali N., Junaini S. N., Mit E., Shiang C. W., Saeed S. Visualisation of user stories in UML models: A systematic literature review. *Acta Inform. Pragensia*. 2023. № 12 (2). Pp. 419–438.
3. Indrayana I. N. E., Sutawinaya P., Pratiwi N. M. W. D., Prihatini P. M., Asri S. A. Android-based child monitoring application using a smartwatch and geofence service. *Journal of Physics: Conference Series*. 2021. № 1803 (1), 012024.
4. Sudarmilah E., Saputra D. B., Arbain A. F. B., Murtiyasa B. Web-based system for growth and development monitoring early childhood. *Journal of Physics: Conference Series*. 2021. № 1874 (1), 012024.
5. Shakhovska N. B., Lytvyn V. V. Proektuvannia informatsiinykh system: navchalnyi posibnyk. Lviv: «Magnoliia-2006», 2011. 380 s.

doi: 10.32403/1998-6912-2025-1-70-35-41

FORMALIZATION OF INFORMATION SYSTEM DESIGN PROCESSES USING UML

A. V. Kudriashova, Yu. O. Biletskyi

*Lviv Polytechnic National University,
12, Stepana Bandery, St., Lviv, 79013, Ukraine
alona.v.kudriashova@lpnu.ua*

A formalized representation of the structure and functionality of an information system for child development monitoring has been carried out using UML. Four types of diagrams are considered: use case, class, activity, and sequence diagrams. The constructed use case diagram reflects the logic of user actions — one of the parents in-

teracting with the mobile application in five main scenarios: monitoring physical activity, maintaining a food diary, recording memories, generating predictions, and viewing history. Each scenario is executed autonomously, without the involvement of external systems. The class diagram provides a detailed view of the system's structure. It includes data classes, DAO components, the ViewModel layer, and the interface. Integration of the TensorFlow Lite module is implemented through a separate class that receives input data from the PredictionViewModel. The types of relationships between classes are defined according to the logic of interaction. The activity diagram captures the internal sequence of user request processing. Initialization of the database and sensors occurs immediately after the application is launched. Subsequently, the user inputs nutritional data, creates textual and visual records, activates the personalized recipe module, and accesses stored history. The sequence diagram demonstrates the dynamic interaction between system components. It illustrates the transmission of requests between the user, interface, ViewModel, and data modules, with particular emphasis on the prediction stage. The set of developed models enables a clear understanding of the system's logic and structure, ensures consistency among components, and facilitates the implementation and testing of the software.

Keywords: UML, use case diagram, class diagram, sequence diagram, activity diagram, modeling, information system, mobile application design.

Стаття надійшла до редакції 20.05.2025.

Received 20.05.2025.