

РОЗРОБЛЕННЯ ПРОТОТИПУ МОБІЛЬНОГО НАВЧАЛЬНОГО ЗАСТОСУНКУ

Котенко Б. О^{1.}, Шапошнікова О. П^{1.}, Мнушка О. В^{2.}¹Харківський національний автомобільно-дорожній університет²Харківська загальноосвітня школа І-ІІІ ступенів № 154

Анотація. Проаналізовані вимоги до навчального застосунку для дітей молодшого віку, були визначені функціональні вимоги та здійснено їх трасування, складено структуру керування застосунком і статичну структуру програми. Вибрано та визначено засоби для реалізації програми, а також досліджено реалізацію програми.

Ключові слова: комп'ютерна програма, мобільна платформа, застосунок для розвитку.

Вступ

Інформаційні технології позитивно впливають на розвиток та навчальний процес дітей. Сучасних дітей, які живуть в епоху цифрових технологій, необхідно підготувати до їх використання. На розвиток дітей дошкільного віку суттєво впливає навколишнє середовище. За умови правильного використання технології можуть стати інструментом навчання дітей дошкільного віку. Важливою є та обставина, що технології ефективніші для навчання, коли дорослі взаємодіють з маленькими дітьми.

Ще один аспект, який покращує повсякденні методи навчання, – це використання його ігрових інструментів. Під час навчання на основі гри діти із задоволенням вільно роблять власний вибір, а також використовують такі технології як інструмент для розвитку своєї творчості.

З огляду на всі передумови щодо тематики розроблення навчального програмного забезпечення важливим є коректне й точне визначення потреб користувача та функціональних вимог, а також для забезпечення необхідної простоти потрібним є створення лаконічної і зрозумілої структури програми.

Для вирішення цього завдання доцільно використовувати нотацію UML (Unified Modeling Language), що є мовою графічного опису для об'єктно-орієнтованого моделювання в галузі розроблення програмного забезпечення.

У межах нашого дослідження необхідно здійснити аналіз потреб потенційних користувачів, розробити поведінкові та статичні моделі застосунків та реалізувати MVP (Minimum Viable Product) застосунку із зазначеної тематики.

Аналіз публікацій

З огляду на поширення навчальних інформаційних технологій у житті дітей важливо

визначити освітній потенціал зазначених застосунків для дітей різного віку. Знання про когнітивні та соціальні здібності дітей стають основою для досвіду розроблення та використання застосунків [1]. У [2] досліджується підхід до використання застосунків для навчання дітей дошкільного віку, зокрема дорослому рекомендується спрямувати взаємодію дитини з цифровими медіа: звертати увагу дитини, допомагати їй регулювати свою поведінку та емоції та співвідносити вміст екрана зі своїм життєвим досвідом.

У [3] проаналізований матеріал про використання об'єктно-орієнтованого підходу під час розроблення навчальних застосунків за допомогою навчальних програм і їхнього дизайну.

У [4] здійснено аналіз популярності операційних систем, вивчені вікові межі навичок дітей, розглянуто матеріал за вибором кольорних палітр, здійснено порівняльний аналіз засобів розроблення та метод розроблення початкового прототипу кросплатформного застосунку.

А також проаналізовано досвід впровадження інформаційних технологій для побудови навчальних застосунків [5].

Мета та постановка завдання

Метою роботи є визначення потреб користувачів застосунку, розроблення поведінкових та статичних моделей, складання його структури, визначення засобів, які будуть використовуватися для реалізації MVP і складання опису основного набору методів, необхідних для функціонування застосунку.

Розроблення моделі прецедентів

Для визначення початкових функціональних вимог було розроблено 2 моделі: модель прецедентів як форма створення вимог (use

case diagram) та модель вимог. Ці дві моделі фактично забезпечують дві «бази даних» функціональних вимог [6, 7].

У процесі моделювання прецедентів визначено межі системи, визначено акторів та прецеденти і відносини між ними. Під час розроблення моделі прецедентів було використано принципи, зазначені в [6–8].

У межах нашого застосунку визначаємо 2 акторів: «Суб'єкт навчання» – дитина дошкільного віку, який буде брати участь в опитуванні (далі – «дитина»), і «Той, хто контролює навчання» – батьки або педагог, які будуть налаштовувати застосунок і переглядати результати (далі – «педагог»).

До прецедентів, які взаємодіють між собою, додаємо відносини. Відносини «include» додаємо для прецедентів, які складаються з інших дій, а «extend» – для прецедентів, які їх розширюють.

До сфери діяльності застосунку додаємо всі прецеденти і зв'язки між ними.

Результат створення моделі прецедентів наведено на рис. 1.

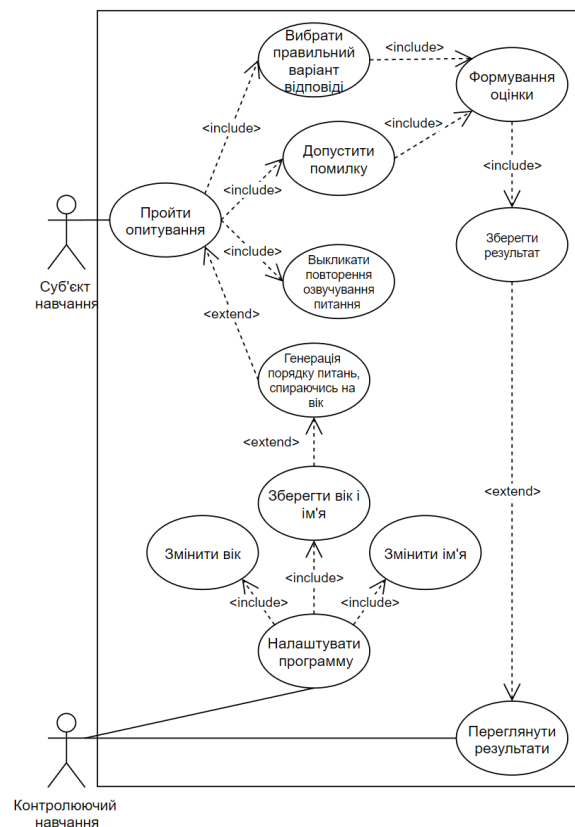


Рис. 1. Use Case діаграма (діаграм прецедентів)

Модель вимог до застосунку

Для розроблення моделі вимог застосунку було використано такий підхід: визначені бізнес-вимоги (business requirement – BR),

після їхнього аналізу визначені вимоги користувача (user requirement – UR), розроблені функціональні вимоги (functional requirements – FR) та класифіковані для їх трасування.

Під час формулювання вимог звертаємо увагу на те, що згідно з BABOK Guide вимога будь-якого рівня має фокусуватися на користі, тобто на основній її функції [8, 9].

BR1 – потреба у можливості застосунку підлаштовуватися під користувача. BR задовольняє UR:

UR1 – педагог потребує можливості внесення до програми даних користувача, який буде здійснювати навчання. UR формує декілька FR:

FR1 – застосунок має реалізовувати збереження даних користувача в базі даних;

FR2 – застосунок має зберігати дані користувача, які складаються з імені та віку для здійснення адаптації.

BR2 – потреба в реалізації лаконічного опитувально-навчального процесу. BR задовольняє декілька UR:

UR1 – дитина хоче легко та швидко орієнтуватися в процесі відповіді на запитання. Ця UR формує декілька FR:

FR1 – застосунок має реалізувати випадкову адаптивну генерацію відповідно до віку дитини;

FR2 – застосунок має реалізувати перехід між запитаннями;

FR3 – застосунок має реалізувати зниження оцінки під час вибору неправильного варіанта відповіді; FR4 – застосунок має реалізувати збереження результату в базі даних; FR5 – застосунок складається з максимально простого інтерфейсу для відповіді; FR6 – застосунок має реалізувати набір тестових запитань з огляду на вік дитини.

UR2 – дитина хоче добре сприймати навчальну гру та концентруватися на опитуванні. UR формує декілька FR:

FR1 – застосунок має реалізувати яскраві та привабливі зображення, які використовуються в опитуванні;

FR2 – застосунок має реалізувати примітивні анімаційні елементи; FR3 – застосунок має здійснювати програвання фонові музики.

BR3 – потреба внести до застосунку персонажа-асистента, який має спростити процес навчання дітей. BR задовольняє декілька UR:

UR1 – дитина хоче мати можливість не тільки прочитати запитання, але й почути його. UR формує декілька FR:

FR1 – застосунок має реалізувати озвучування та можливість повторення запитань; FR2 – застосунок має реалізувати чітке озвучування без переривань.

UR2 – Дитина хоче отримувати реакцію персонажа-асистента під час вибору варіанта відповіді. UR формує декілька FR:

FR1 – застосунок має реалізувати схвальну фразу під час вибору правильного варіанта відповіді; FR2 – застосунок має реалізувати звукове прохання до дитини спробувати ще раз, якщо була надана помилкова відповідь.

BR4 – потреба зберігати результати роботи з користувачем. BR задовольняє декілька UR:

UR1 – педагог хоче мати можливість переглядати результати користувача, який буде здійснювати процес навчання. UR формує декілька FR:

FR1 – застосунок має реалізувати збереження результатів в базі даних та їх відображення;

FR2 – застосунок має реалізувати збереження оцінки, визначити дату та час, вік та ім'я дитини.

UR2 – педагог хоче мати можливість видаляти непотрібні результати. UR формує декілька FR:

FR1 – застосунок має реалізувати можливість вибрати конкретний результат для видалення;

FR2 – застосунок має реалізувати можливість видалення вибраного результату з бази даних і з форми застосунку.

Як було зазначено, обидві моделі забезпечують дві «базы даних» функціональних вимог. Для запобігання їхніх розбіжностей у цих базах, щоб з'ясувати наявність чогось, що не охоплено іншою або навпаки, було здійснено процес співставлення. Таке визначення питання є одним з аспектів відображення вимог.

Статична модель застосунку

Статична структура застосунку є сукупністю всіх методів, функцій, структур і даних, які будуть повноцінно функціонувати. Для вирішення цих питань найбільш оптимальною є діаграма класів.

З огляду на визначені особливості програмного забезпечення складаємо діаграму класів [11–12] та вносимо до неї всі необхідні дані, методи й обробники подій.

Під час складання діаграми класів була використана інформація з [9–12]. Результат наведено на рис. 2, 3.

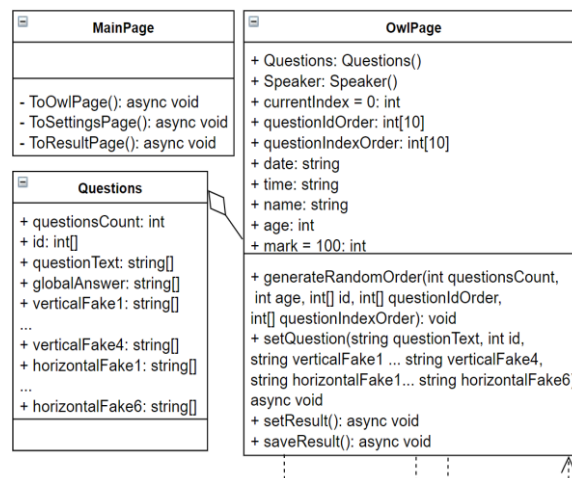


Рис. 2. Діаграма класів застосунку (1)

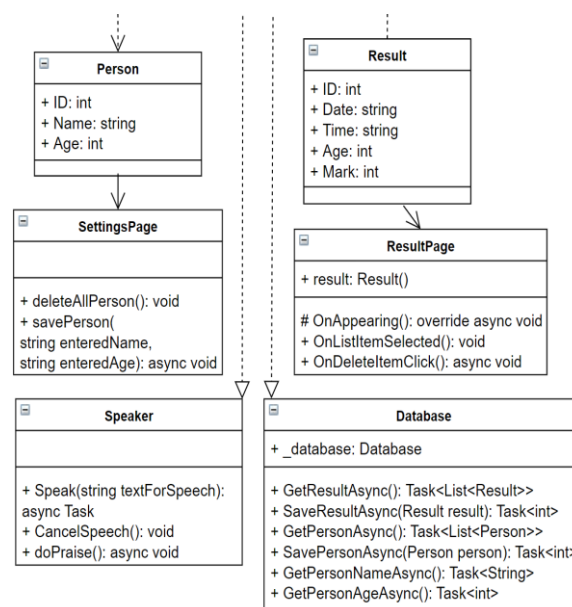


Рис. 3. Діаграма класів застосунку (2)

Розроблення застосунку

Під час розроблення програми використовуємо структуру, створену за допомогою діаграми класів (рис. 2–3), а для реалізації потреб використовуємо додаткові пакети NuGet. Під час цього процесу були враховані рекомендації з розроблення програмного забезпечення [13–15].

Оскільки повний аналіз програми є надмірно великим і недоцільним, то в межах роботи розглянемо тільки основні частини створеного застосунку.

У програмі, що розроблено, створено 4 форми, які прив'язані до класів:

- MainPage – початкова навігаційна сторінка застосунку, містить 3 кнопки для переходу до інших форм застосунку;

- SettingsPage – сторінка, що містить елементи для внесення та відображення збережених у базі даних імені та віку користувача;

- OwlPage – основна частина програми, в якій здійснюється генерація порядку запитань, складність яких залежить від зазначеного в класі SettingsPage віку, їх відображення (за кодом запитання), його озвучення та збереження результату в базі даних.

- ResultPage – сторінка, на якій подано список результатів (дата, час, вік, бали) і кнопка для видалення вибраного результату з бази даних застосунку.

Інтерфейс цих форм складено за допомогою мови XAML.

Для збереження даних користувача (структура Person) і результатів опитування (структура Result) був створений клас Database, що використовує методи з пакета sqlite, який містить асинхронні (async) функції формату <Task> для збереження даних користувача і результатів опитування, а також для повернення значень з бази даних.

Запитання розташовані в структурі Questions у текстовому і чисельному форматах. Залежно від значень у випадковому сгенерованому порядку запитань в класі OwlPage залежить, які елементи інтерфейсу будуть завантажуватися і відображатися.

Клас Speaker здійснює озвучування тексту запитань, їх повторення і коментарі щодо коректності вибраної відповіді. Крім того, у класі за допомогою асинхронності методів реалізовано дотримання порядку озвученого тексту.

Також, в основній частині програми були використані функції для таймінгу «Task.Deley ()», завдяки яким були реалізовані затримки між відображеннями частин інтерфейсу.

Ця реалізація, що є робочим MVP-прототипом мобільної програми, може бути оптимізована в подальшій роботі.

Висновки

У роботі здійснено аналіз потреб користувачів і були визначені функції навчального застосунку. На відміну від [4] здійснено аналіз вимог та побудовані поведінкові моделі: діаграма прецедентів та модель вимог. Для визначення правильності завдання і реалізації вимог до застосунку на процесах життєвого циклу було здійснено трасування вимог, що дозволило

врахувати потреби користувача та відповідні зміни в прототипі програмного забезпечення з метою підвищення його якості та юзабіліті.

Для презентації статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування була розроблена статична модель – діаграма класів.

На основі складених моделей було створено прототип програми, що містить MVP застосунку. Перспективами подальших досліджень є вдосконалення методів та алгоритмів подання навчального контенту з огляду на потреби цільової аудиторії користувачів.

Література

1. Learning in the Digital Age: Putting Education Back in Educational Apps for Young Children / Zosh JM, Hirsh-Pasek K, Michnick Golinkoff R, Parish-Morris J, Tremblay RE, Boivin M, Peters RDeV, eds. Rvachew S, topic ed. Encyclopedia on Early Childhood Development [online]. <https://www.child-encyclopedia.com/technology-early-childhood-education/according-experts/learning-digital-age-putting-education-back>. Published: November 2016. Accessed October 1.
2. Rvachew S. Technology in early childhood education: overall commentary. In: Tremblay RE, Boivin M, Peters RDeV, eds. Rvachew S, topic ed. Encyclopedia on Early Childhood Development [online]. <https://www.child-encyclopedia.com/technology-early-childhood-education/according-experts/technology-early-childhood-education-overall>. Published: November 2016. Accessed October 2. 2022.
3. Котенко Б. О., Мнушка О. В. Об'єктно-орієнтований підхід до дизайну навчаючих програм. Комп'ютерні технології і мехатроніка: зб. наукових праць за матеріалами Міжнародної науково-практичної конференції. Харків: ХНАДУ, 2019. С. 125–127.
4. Мнушка О. В., Котенко Б. О., Савченко В. М. Аналіз вимог та розробка прототипу навчаючого програмного забезпечення для мобільних платформ. Вісник Харківського національного автомобільно-дорожнього університету. 2021. Вип. 92. Т. 1. С. 51–59.
5. Шапошнікова О. П., Рудакова С. С. Використання інформаційних технологій у методиці викладання дисципліни «Комп'ютерна електроніка». Автомобільний транспорт. 2012. Вип. 31. С. 109–114.
6. Martin Fowler. UML Distilled: a Brief Guide to the Standard Object Modeling Language. Publisher: «Addison-Wesley Professional». 2016. P. 208.
7. A Guide to the Business Analysis Body of Knowledge (BABOK Guide). Publisher: International Institute of Business Analysis. 2015. P. 512.

8. Лантрат О. В., Сахно Є. В., Шапошнікова О. П. Розроблення мобільного додатку «Міські парковки». Вісник ХНАДУ. 2019. Вип. 87. С. 59–66.
9. Лантрат О. В., Сахно Є. В., Шапошнікова О. П. Архітектура додатку для пошуку місць для паркування. Вісник ХНАДУ. 2020. Вип. 89. С. 7–15.
10. Модификаторы доступа. URL: <https://metanit.com/sharp/tutorial/3.2.php>.
11. Lee R.Y. Object-Oriented Software Engineering With UML: A Hands-on Approach (Computer Science, Technology, and Applications). Nova Science Pub Inc. 2019. 375 p.
12. Діаграми класів. URL: https://uk.wikipedia.org/wiki/Діаграма_класів.
13. Grady B. Object-Oriented Analysis and Design with Applications. Addison-Wesley 2. 2007.
14. C. Petzold. Creating Mobile Apps with Xamarin.Forms, 2016. URL: <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/creating-mobile-apps-xamarin-forms/>.
15. Руководство по программированию для Xamarin Forms. URL: <https://metanit.com/sharp/xamarin/>.
6. Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language. Publisher: «Addison-Wesley Professional». 2016. P 208.
7. A Guide to the Business Analysis Body of Knowledge (BABOK Guide). Publisher: International Institute of Business Analysis. 2015. P. 512.
8. Lantrat O. V., Sakhno Y. V., Shaposhnikova O. P. Development of the "City Parking" mobile application. Visnyk KhNADU. 2019. Vyp. 87. P. 59–66.
9. Lantrat O.V., Sakhno Y. V., Shaposhnikova O. P. Architecture of the application for finding parking places. Visnyk KhNADU. 2020. Vyp. 89. P. 7–15.
10. Access modifiers. URL: <https://metanit.com/sharp/tutorial/3.2.php>.
11. Lee R. Y. Object-Oriented Software Engineering With UML: A Hands-on Approach (Computer Science, Technology, and Applications). Nova Science Pub Inc. 2019. 375 p.
12. Class Diagrams. URL: https://uk.wikipedia.org/wiki/Діаграма_класів.
13. Grady B. Object-Oriented Analysis and Design with Applications. Addison-Wesley 2. 2007.
14. C. Petzold. Creating Mobile Apps with Xamarin.Forms, 2016. URL: <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/creating-mobile-apps-xamarin-forms/>.
15. Xamarin Forms Programming Guide. URL: <https://metanit.com/sharp/xamarin/>. [In Russian].

References

1. Learning in the Digital Age: Putting Education Back in Educational Apps for Young Children / Zosh JM, Hirsh-Pasek K, Michnick Golinkoff R, Parish-Morris J. In: Tremblay RE, Boivin M, Peters RDeV, eds. Rvachew S, topic ed. Encyclopedia on Early Childhood Development [online]. <https://www.child-encyclopedia.com/technology-early-childhood-education/according-experts/learning-digital-age-putting-education-back>. Published: November 2016. Accessed October 1.
 2. Rvachew S. Technology in early childhood education: overall commentary. In: Tremblay RE, Boivin M, Peters RDeV, eds. Rvachew S, topic ed. Encyclopedia on Early Childhood Development [online]. <https://www.child-encyclopedia.com/technology-early-childhood-education/according-experts/technology-early-childhood-education-overall>. Published: November 2016. Accessed October 2, 2022.
 3. Kotenko B. O., Mnushka O. V. Object-oriented approach to design teaching programs. Computer Technology and Mechatronics. Proceedings of the International scientific-practical conference. 2019. PP. 125–127 [In Ukrainian].
 4. Mnushka O. V., Kotenko B. O., Savchenko V. M. Analyzing requirements and developing prototype of training software for mobile platforms. Bulletin of Kharkov National Automobile and Highway University. 2021. Issue 92 (1). P. 51–59 [In Ukrainian] DOI: 10.30977/BUL.2219-5548.2021.92.1.51.
 5. Shaposhnikova O. P., Rudakova S. S. The use of information technologies in the teaching methodology of the discipline "Computer Electronics". Automobile transport. 2012. Vyp. 31. P. 109–114.
- Котенко Богдан Олександрович**, студент, кафедра комп'ютерних технологій і мехатроніки, тел. (+38099)328-68-70, e-mail: bohdan.kotenko.1997@gmail.com, Харківський національний автомобільно-дорожній університет, вул. Ярослава Мудрого, 25, м. Харків, 61002, Україна;
- Шапошнікова Олена Павлівна**, к.т.н., доцент кафедри комп'ютерних технологій і мехатроніки, тел. +38-057-707-37-43, e-mail: shaposhnikovaep@gmail.com, Харківський національний автомобільно-дорожній університет, вул. Ярослава Мудрого, 25, м. Харків, 61002, Україна;
- Мнушка Оксана Василівна**, вчитель інформатики, тел. (+38057)725-85-53, e-mail: mnushka.ov@gmail.com, Харківська загальноосвітня школа І-ІІІ ступенів № 154 Харківської міської ради Харківської області, пр. Л. Свободи, 42Б, м. Харків, 61204, Україна.

Developing a mobile educational application prototype

Abstract. Problem. Over the past few decades, the use of information and computer technology has expanded exponentially. In many fields of activity

during this period, various software tools were integrated, and some of the fields were almost completely automated. For educational software development, it is crucial to correctly and accurately identify needs, draw up requirements, and consider the necessary simplicity, creating a concise and understandable program structure. **Goal.** The purpose of this work is to identify the needs of application users, formulate requirements for the application, create a program model, draw up its structure, determine the means that will be used for its implementation and compose a description of the main set of methods necessary for the function-analysis of a developing application. **Methodology.** Analytical research methods, requirements elicitation, application structuring using UML diagrams, and teaching software development methods. **Results.** An analysis of the needs of users of the learning application for young children was carried out, the requirements were collected and traced, and the program's application control structure and the static structure of the program were drawn up. The means for the implementation of the program were selected and described, and the implementation of the program was described. **Originality.** Requirements, structure, and approach to creating an appli-

cation are based on the analysis of children's knowledge levels and needs. **Practical value.** Such software can be adapted for different knowledge levels and various topics unrelated to children's education.

Key words: computer program, mobile platform, development application.

Kotenko Bohdan, student, Department of Computer Technologies and Mechatronics,

tel. (+38099)328-68-70,

e-mail: bohdan.kotenko.1997@gmail.com;

Shaposhnikova Olena, candidate of technical sciences, Department of Computer Technology and Mechatronics, tel. +38-057-707-37-43,

e-mail: shaposhnikovaep@gmail.com, Kharkiv National Automobile and Highway University, 25, Yaroslava Mudrogo str., Kharkiv, 61002, Ukraine;

Mnushka Oksana, Computer Science teacher,

tel. (+38057)725-85-53,

e-mail: mnushka.ov@gmail.com, Kharkiv Secondary School of Grades I-III No. 154 of the Kharkiv City Council of the Kharkiv Region, 61204, Ukraine, Kharkiv, L. Svobody Ave., 42B.
