

ПРОТОТИП ЗАСТОСУНКУ ГОЛОСОВОГО ПОМІЧНИКА ЛІКАРЯ

Юрій Кіш¹, Сергій Пасічник², Наталія Кунанець², Василь Кут¹

¹ Ужгородський національний університет,
кафедра інформатики та фізико-математичних дисциплін, Ужгород, Україна

² Національний університет “Львівська політехніка”,
кафедра інформаційних систем та мереж, Львів, Україна
E-mail: kish.yurii@uzhnu.edu.ua, ORCID: 0009-0000-6167-0129
E-mail: serhii.o.pasichnyk@lpnu.ua, ORCID: 0009-0006-8011-5618
E-mail: nataliia.e.kunanets@lpnu.ua, ORCID: 0000-0003-3007-2462
E-mail: vasilij.kut@uzhnu.edu.ua, ORCID: 0000-0001-5267-331X

© Кіш Ю. В., Пасічник С. О., Кунанець Н. Е., Кут В. І., 2023

У дослідженні розглянуто основні підходи до візуалізації та моделювання ключових процесів розроблення програмного забезпечення у домені телемедицини, а саме такого застосування, як голосовий помічник лікаря. Інформаційне моделювання здійснено з використанням UML-діаграм, проаналізовано особливості різних етапів життєвого циклу розроблення софту, зокрема сформульовано вимоги до продукту, розглянуто процедуру проєктування, описано основний функціонал застосування, технології фронтенду тощо. За допомогою концептуального моделювання продемонстровано функціонал інформаційної системи – прототипу застосування голосового помічника лікаря. Проаналізовано підходи до використання ефективної frontend технології для застосування.

Ключові слова: моделювання; діаграма класів; діаграма активностей; діаграма прецедентів; голосовий помічник лікаря; frontend.

Постановка проблеми

Налагодження своєчасної медичної допомоги у сучасному суспільстві ґрунтується на технологіях дистанційного консультування. Це зумовило розвиток інноваційного напрямку медицини, що отримав назву телемедицина. Застосування процедур дистанційного консультування зменшує кількість госпіталізованих пацієнтів та завантаження лікарень, сприяє вчасному коригуванню протоколу лікування пацієнтів із хронічними захворюваннями. Разом з тим, телемедицина стає важливим складовим елементом змін у системі охорони здоров'я, оскільки продемонструвала свої безпосередні переваги, даючи змогу мільйонам пацієнтів продовжувати лікування без стаціонарного перебування в лікарні. Це стало особливо відчутним в умовах пандемії. Запровадження дистанційного обстеження – ефективний спосіб зменшити потік пацієнтів у медичних закладах, не залишаючи поза увагою пацієнтів, що потребують постійного супроводу лікаря.

У первинній медичній допомозі телемедицина зазвичай реалізується у формі телефонних дзвінків, коли пацієнт звертається за порадою до лікаря щодо медичних проблем, для вирішення яких не потрібно відвідувати лікаря. Телемедицина не замінює процедуру консультування, а переводить її на інший рівень. Роль телемедицини полягає у наданні медичних послуг у зручному для пацієнта форматі, сприяє комунікації лікаря та пацієнта, звільняючи його від фізичного візиту для отримання медичної консультації або призначення лікування.

Саме тому чимало вітчизняних стартапів націлено на створення продукту в цьому домені. Хоча, безперечно, орієнтуються вони, передусім, на міжнародний ринок. Продукт компанії повинен відповідати певним стандартам, щоб бути конкурентоспроможним у світі. Для цього потрібно мати

чітке уявлення, що і як необхідно робити на кожному етапі створення програмного забезпечення. Візуалізація архітектури інформаційної системи, концептуальне моделювання основного функціоналу істотно поліпшують розуміння її роботи, а також сприяють забезпеченню якості програмного продукту на належному рівні. UML діаграми є зручним інструментарієм, що дає змогу реалізувати візуалізацію та моделювання робіт з аналізу вимог до застосунку, його проектування.

Аналіз останніх досліджень

Як свідчать результати дослідження науковців Гарвардського університету, в медичній галузі США 2020 р. вирізняється зменшенням кількості пацієнтів, що записалися на амбулаторний прийом, надавши перевагу отриманню дистанційних консультацій. Попит на дистанційні медичні послуги поступово зростає, що потребує інвестування в розвиток телемедицини. Згідно із прогнозами Global Market Insights, світовий ринок телемедицини досягне до 2027 р. \$187,7 млрд. У 2019 р. він оцінювався в \$45,5 млрд, у 2020 р. – в \$55,9 млрд, а у 2021 р. – \$72,7 млрд [3].

UML діаграми – зручний інструмент у професійному наборі засобів, якими послуговуються проєктні менеджери, бізнес-аналітики, девелопери, інженери із забезпечення якості продукту. Є чимало публікацій, які стосуються цієї мови моделювання. Проте потрібно звернути увагу насамперед на ті наукові розвідки, у яких акцентують на практичному застосуванні діаграм.

На відміну від класичного набору інструментів, множина UML діаграм складається з 14 різновидів, які умовно можна поділити на дві групи: структурні діаграми та діаграми поведінки. В цьому дослідженні використано можливості практичного застосування трьох з них: діаграми класів (Class diagram), діаграми активностей (Activity diagram), а також діаграми прецедентів (Use-case diagram) [7].

Діаграма класів – це UML-діаграма, яка описує систему, візуалізуючи різні типи об'єктів усередині системи та види статичних зв'язків, що існують між ними. Вона також ілюструє операції та атрибути класів. Зазвичай її використовують для вивчення концепцій галузі, розуміння вимог до програмного забезпечення та описання детальних проєктів [2, 4].

Діаграма активностей візуалізує поведінку розроблюваної інформаційної системи у вигляді блок-схеми, що відображає її бізнес-процеси, логічні зв'язки процедур, притаманні інформаційній системі, та потоки робіт, які здійснюються в ній. По суті, це моделювання логіки поведінки системи чи взаємодії кількох систем [1, 6].

Діаграма прецедентів – це тип поведінкової діаграми UML, що часто використовується для аналізу різних систем. Вона дає змогу візуалізувати різні типи акторів, які користуватимуться функціями інформаційної системи, і те, як вони можуть взаємодіяти із інформаційною системою. Діаграма відображає основні функції розроблюваної інформаційної системи, якими може послуговуватися кожен із користувачів, а також активно використовується для верифікації роботи програмного продукту. Це зручний інструмент для розроблення тестових сценаріїв – test case. “Юзкейси, за замовчуванням, є тестованими вимогами із зазначеною метою і шляхом її досягнення” [3, 5].

Формулювання цілі статті

Це дослідження має на меті проаналізувати особливості застосування діаграм UML для моделювання та візуалізації основного функціоналу прототипу голосового помічника лікаря.

Виклад основного матеріалу

Для аналізу практичного застосування ми вибрали прототип стартапу “Huckleberry” (голосовий помічник лікаря). “Huckleberry” дає змогу автоматизувати документацію типових активностей лікаря, таких як: розмови під час огляду пацієнта, встановлення діагнозу, запис пацієнта на прийом, заповнення медичної картки, історії хвороби пацієнта тощо, дистанційного консультування за допомогою технологій доповненої реальності та машинного навчання.

Голосовий помічник відповідає усім сучасним вимогам до програмних продуктів у медичній галузі. Це стосується як суто технічної частини, так і стандартів якості та безпеки. На ринку України аналогічних продуктів немає. Якщо говорити про найкращі світові стартапи в цій галузі, то завдяки такому додатковому функціоналу, як, наприклад, інтеграція з лабораторіями, можливість проведення

групових консультацій, сапорт щодо перевірки коректності медичної термінології та призначень, то цей продукт буде конкурентним і на світовому ринку.

Проаналізуємо етап життєвого циклу проекту, спрямованого на розроблення інформаційної системи. Для прототипу інформаційної системи використаємо діаграму класів. Типовий алгоритм дій такий: визначаємо класи системи та їх атрибути, встановлюємо відношення між класами, і тільки після цього переходимо до побудови діаграми класів.

Таблиця 1

Типові класи системи “Huckleberry”

Класи		Атрибути		Методи	
Назва	Призначення	Назва	Зміст	Назва	Зміст
Person	Основний актор	name	ім'я	create_doc	створення документа
		email	е-мейл	edit_doc	редагування документа
		salary	зарплата	delete_doc	видалення документа
Doctor	Основні активності лікаря	md_spec	спеціалізація лікаря	sign_note	підпис картки
		coop_form	форма співпраці з клінікою	transf_note	трансфер картки в EMR
Nurse	Основні активності медсестри	nurs_spec	спеціалізація медсестри	sched	операції із розкладом
Docs	Різні види документів	id	ідентифікатор документа	select_type	вибрати тип
		int_struct	структура документа	create_id	створити документ і присвоїти номер
		note_type	тип документа	send_emr	трансфер документа в EMR систему
		emr_integr	можливість перенесення в EMR		
Create Note	Створення документа	Boolean	створено/Ні		
Note Verification	Верифікація документа	med_term	верифікація медичних термінів	auth_note	авторизація
		emr_ver	верифікація кодів хвороб		
Note Transferring to EMR	Трансфер документа в EMR	note_id	ідентифікатор документа		
		note_md	ідентифікатор лікаря		
		note_cont	зміст документа		

Встановимо відношення між визначеними класами (табл. 2).

Таблиця 2

Відношення між класами системи “Huckleberry”

Класи		Вид відношення	Розмірність
Person	Doctor	Inheritance	1...*
Person	Nurse	Inheritance	1...*
Person	Docs	Association	1...*
Docs	Create Note	Realization Implementation	1...*
Docs	Note Transferring to EMR	Realization Implementation	1...*
Create Note	Note Verification	Dependency	1...*

Далі переходимо до проектування архітектури системи і будемо діаграму класів (рис. 1).

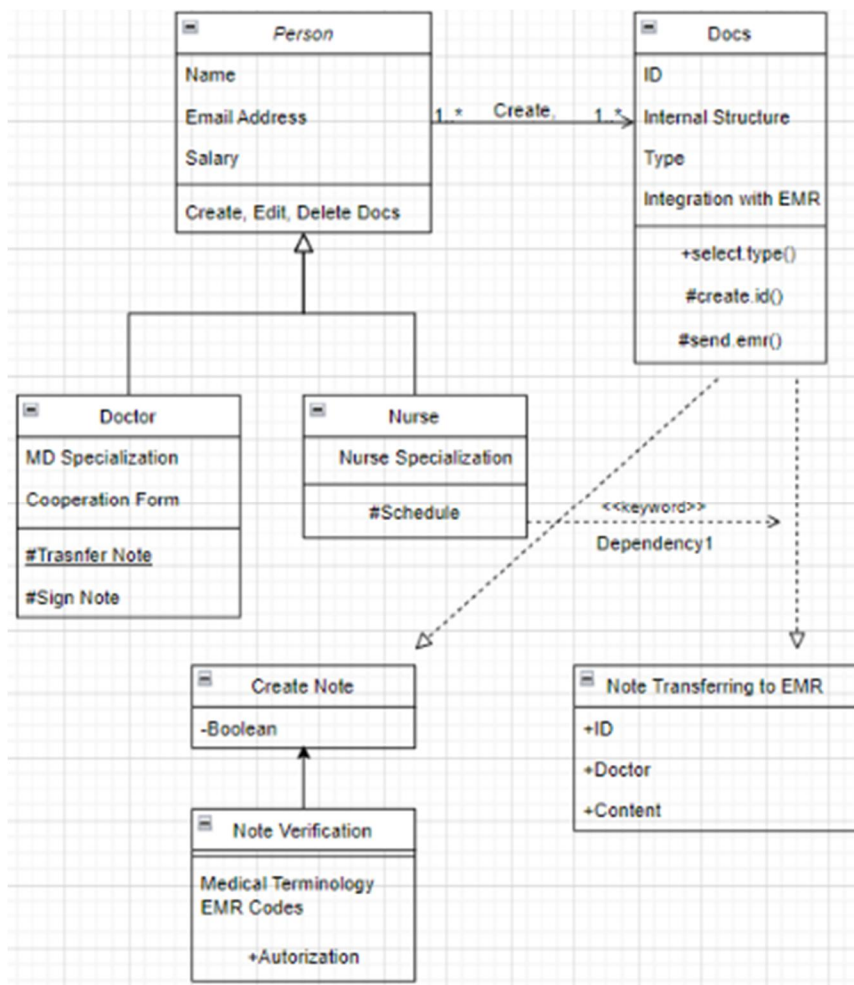


Рис. 1. Діаграма класів системи “Huckleberry”

На наступному кроці переходимо до моделювання поведінкової активності системи. Для прикладу візьмемо типовий флов поведінки системи під час створення *Картки прийому пацієнта* (рис. 2).

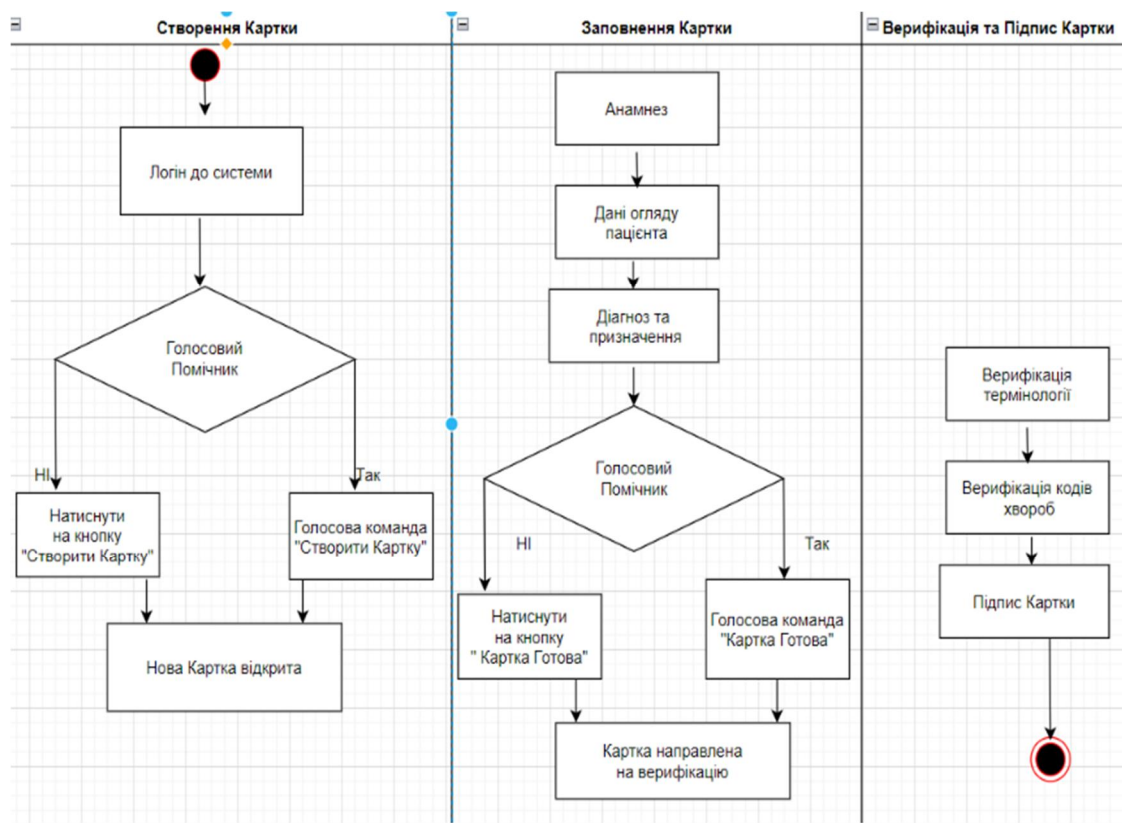


Рис. 2. Робота системи "Huckleberry" під час створення Картки прийому пацієнта

Здійснимо візуалізацію функціонування системи у разі додавання нової хвороби до історії хвороби пацієнта (рис. 3).

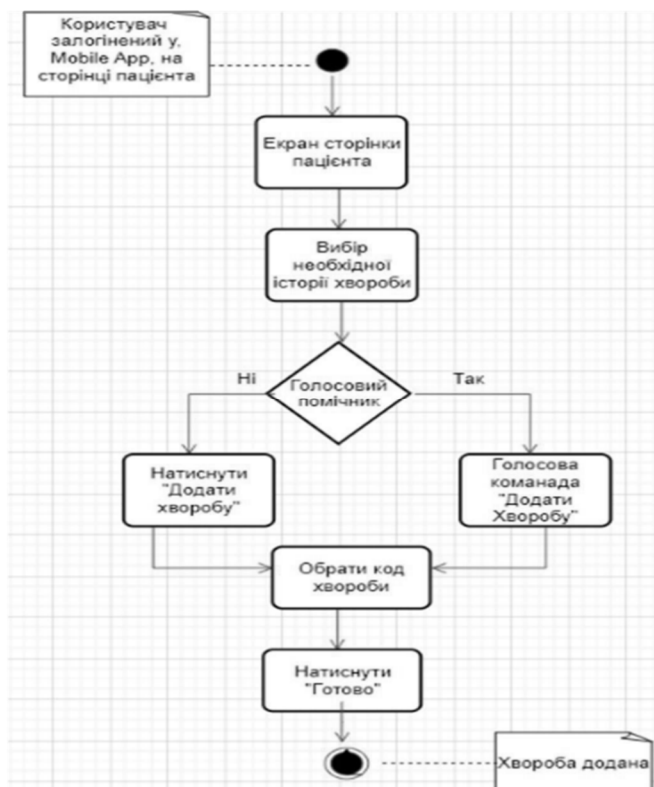


Рис. 3. Функціонування системи "Huckleberry" у разі додавання нової хвороби до історії хвороби пацієнта

Основні функціональні вимоги застосунку “Huckleberry” можна зобразити та протестувати за допомогою діаграми прецедентів (рис. 4).

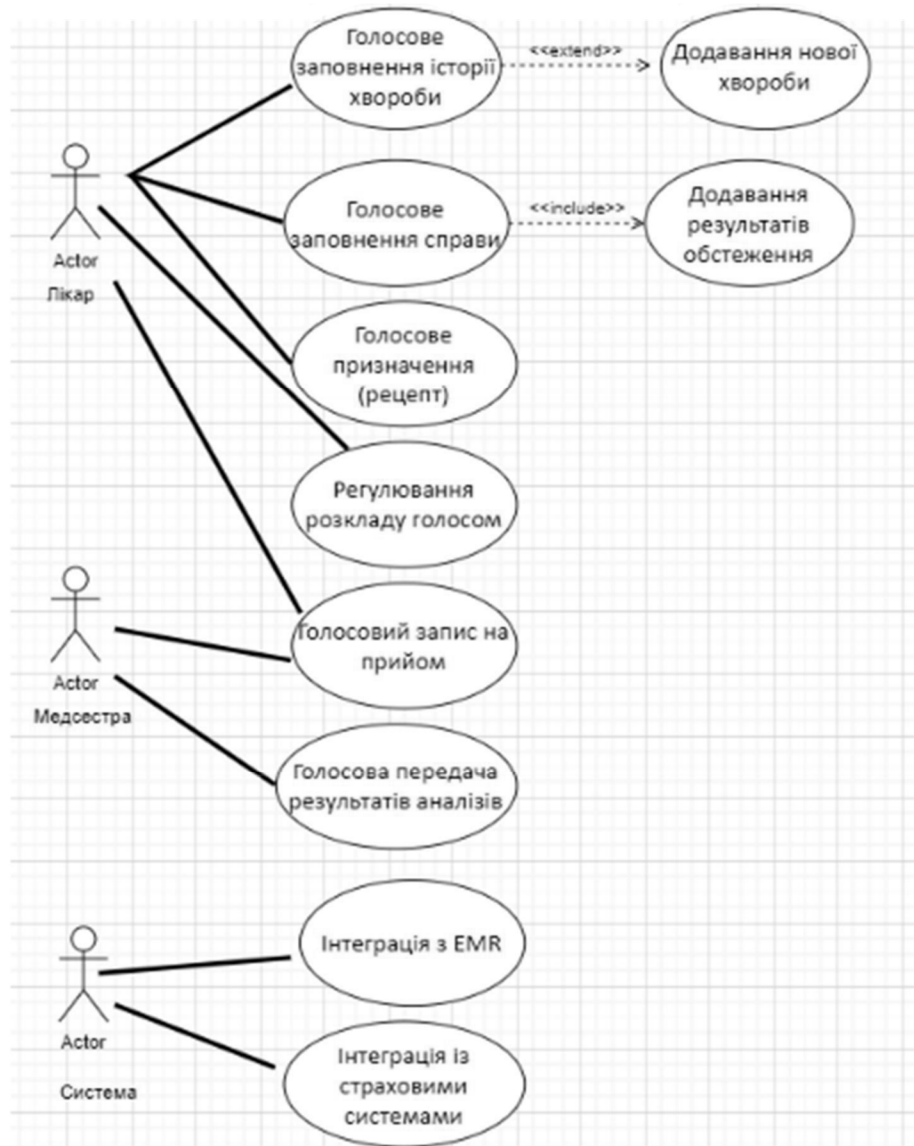


Рис. 4. Функціональні вимоги застосунку “Huckleberry”

Діаграми UML дають змогу всім учасникам розроблення програмного забезпечення: девелоперам, проєкт-менеджерам, бізнес-аналітикам, тестувальникам, а також представникам замовника послуговуватися зрозумілою всім мовою.

Важливе завдання – розроблення зручного інтерфейсу інформаційної системи із використанням ефективної frontend технології [8]. Фронтенд розробка є важливим етапом розроблення будь-якої інформаційної системи та вимагає від розробника знань та досвіду використання різних технологій та підходів до її створення. Якість застосунку істотно залежить від фронтенд технології, яка складається зі створення інтерфейсу користувача, що забезпечує взаємодію користувача з інформаційною системою. Під час розроблення користувальницького інтерфейсу застосунку послуговувались такими критеріями, як вигляд та простота використання, приділяючи велику увагу дизайну та взаємодії з користувачем. Налагоджено тісну співпрацю з дизайнерами, бекенд-розробниками та менеджерами проєктів, щоб переконатися, що кінцевий продукт відповідає потребам клієнта та забезпечує найкращий функціонал для кінцевих користувачів.

Фронтенд розробка інформаційної системи – це процес створення клієнтської частини програмного забезпечення (зазвичай вебзастосунків), яка відображається користувачу через браузер або мобільний застосунок. У фронтенд частині відбуваються проєктування, розроблення і тестування інтерфейсу користувача, забезпечення його взаємодії із бекендом, надійності та безпеки. Розроблення функціональності інтерфейсу – реалізація логіки взаємодії з користувачем, а також забезпечення взаємодії із бекендом, а також легкої навігації та зручного доступу до функцій системи. Користувачі бачать і взаємодіють тільки з тією частиною застосунку, над якою працювали Front-end розробники. Все, що бачить користувач на сайті – від шрифту, фону, меню, що випадає, слайдера, створено за допомогою HTML, CSS і JavaScript, контролюється браузером, який встановлено на комп'ютері користувача. Front-end технологія сприяє створенню “обличчя” застосунку, враховуючи досвід взаємодії користувачів із застосунком. Фронтенд розробка інформаційної системи “голосовий помічник лікаря” забезпечує належну роботу інтерфейсу на різних пристроях та платформах. Під час розроблення інтерфейсу цього застосунку використано комбінацію технологій, з метою створення вебсторінок та застосунку з інтерактивним інтерфейсом. Використання JavaScript дало змогу поліпшити загальну інтерактивність застосунку, моделювати анімовані компоненти інтерфейсу користувача, такі як повзунки зображень, вікна, які спливають, розширені меню навігації тощо. JavaScript дає змогу розширити його функціональність, якої інакше неможливо досягти лише за допомогою HTML і CSS. JavaScript дозволяє реагувати на дії користувачів і динамічно оновлюватися. Завдяки JavaScript цей процес не потребує перезавантаження, щоб змінити її представлення. Так створено адаптивний дизайн інтерфейсу застосунку, що дає змогу використовувати його з мобільних пристроїв, настільних ПК. Адаптивний дизайн дає можливість функціям та вмісту адаптуватися до розміру екрана та апаратного забезпечення. Наприклад, використовуючи застосунок з настільного комп'ютера з великим монітором, користувач отримує численні стовпці, важку графіку, тобто інтерфейс користувача підлаштовується спеціально для миші та клавіатури. На мобільних пристроях інтерфейс відображається як один стовпець, налаштований на сенсорну взаємодію, але з тими самими базовими даними.

Висновки

Телемедицина набула поширення нарівні із наданням традиційних медичних послуг. Запропонований прототип застосунку голосового помічника лікаря “Huckleberry” дає змогу автоматизувати документацію типових активностей лікаря, таких як: розмови під час огляду пацієнта, встановлення діагнозу, запис пацієнта на прийом, заповнення медичної картки, історії хвороби пацієнта тощо, дистанційного консультування за допомогою технологій доповненої реальності та машинного навчання.

Телемедицина не потребує значних витрат, окрім вебкамери та захищеного порталу для пацієнтів, який з'єднує лікаря із захищеною електронною базою даних медичних записів онлайн. Запропонована інформаційна система гарантує безпеку приватної інформації, яка обговорюється під час телемедичного дзвінка, а також надає лікарю-клініцисту можливість зберігати необхідні медичні записи. Забезпечуючи надання онлайн-консультації, застосунок також полегшує роботу лікаря.

З використанням діаграм UML виконано концептуальне моделювання основних активностей інформаційної системи голосового помічника “Huckleberry”, а також верифікацію функціональних вимог. У результаті створено описову, графічну модель застосунку, що дає змогу проаналізувати його роботу із різних ракурсів. Проаналізовано особливості створення інтерфейсу інформаційної системи.

Список літератури

1. Booch G., Jacobson I., Rumbaugh J. (2021). The Unified Modeling Language User Guide, Addison-Wesley, 83–90.

2. Fowler M., Scott K. (2021). UML Distilled: A Brief Guide to the Standard Object Modeling Language. Addison-Wesley, 81–86.
3. Global Market Insights Telemedicine Market Size & Share, Growth Outlook 2021–2027 (2021). URL: <https://www.gminsights.com/industry-analysis/telemedicine-market>.
4. Roman A. (2021). A Study Guide to the ISTQB® Foundation Level Syllabus, Springer, 54–57.
5. Wiegers K., Beatty J. (2021). Software requirements (developer best practices). 3rd edition, 336–346.
6. Авраменко А., Авраменко В. (2021). Проектування інформаційних систем, Черкаси, 111–119.
7. Каграманова Ю. (2022). Як будувати UML-діаграми. Розбираємо три найпопулярніші варіанти. URL: <https://dou.ua/forums/topic/40575/?from=slider>.
8. Artto K. (2015). From the front end of projects to the back end of operations: Managing projects for value creation throughout the system lifecycle. *International Journal of Project Management*, 34(2). DOI: 10.1016/j.ijproman.2015.05.003.

References

1. Booch G., Jacobson I., Rumbaugh J. (2021). The Unified Modeling Language User Guide, Addison-Wesley, 83–90.
2. Fowler M., Scott K. (2021). UML Distilled: A Brief Guide to the Standard Object Modeling Language. Addison-Wesley, 81–86.
3. Global Market Insights Telemedicine Market Size & Share, Growth Outlook 2021–2027 (2021). URL: <https://www.gminsights.com/industry-analysis/telemedicine-market>.
4. Roman A. (2021). A Study Guide to the ISTQB® Foundation Level Syllabus, Springer, 54–57.
5. Wiegers K., Beatty J. (2021). Software requirements (developer best practices). 3rd edition, 336–346.
6. Avramenko A., Avramenko V. (2021). Information system design, Cherkasy, 111–119.
7. Kahramanova Yu. (2022). How to use UML diagrams. We consider three most popular options. URL: <https://dou.ua/forums/topic/40575/?from=slider>.
8. Artto K. (2015). From the front end of projects to the back end of operations: Managing projects for value creation throughout the system lifecycle, *International Journal of Project Management*, 34(2). DOI: 10.1016/j.ijproman.2015.05.003.

DOCTOR'S VOICE ASSISTANT APPLICATION PROTOTYPE

Yurii Kish¹, Serhii Pasichnyk², Nataliia Kunanets², Vasilij Kut¹

¹ Uzhhorod National University,
Department of Informatics and Physical and Mathematical Disciplines, Uzhhorod, Ukraine

² Lviv Polytechnic National University,
Information Systems and Networks Department, Lviv, Ukraine
E-mail: kish.yurii@uzhnu.edu.ua, ORCID: 0009-0000-6167-0129
E-mail: serhii.o.pasichnyk@lpnu.ua, ORCID: 0009-0006-8011-5618
E-mail: nek.lviv@gmail.com, ORCID: 0000-0003-3007-2462
E-mail: vasilij.kut@uzhnu.edu.ua, ORCID: 0000-0001-5267-331X

© Kish Yu. V., Pasichnyk S. O., Kunanets N. E., Kut V. I., 2023

This research examines the main approaches to visualization and modeling of key processes in the creation and testing of software in the domain of telemedicine, namely, such an application as a voice assistant to a doctor. Emphasis is placed on the use of UML diagrams at various stages of the software development life cycle, including requirements analysis, design, description of basic functionality, documentation, testing, etc. With the help of modeling, the answer not only to the questions “what does the system do” and “what does the system consist of” was demonstrated, but also to “how and how efficiently the system works” – in this case, a prototype of a doctor’s voice assistant application. Approaches to using effective frontend technology for the application are analyzed.

Key words: modeling; Class diagram; Activity diagram; Use case diagram; doctor voice assistant.