

Міністерство освіти і науки України  
Кам'янець-Подільський національний університет імені Івана Огієнка  
Фізико-математичний факультет  
Кафедра інформатики

Дипломна робота  
магістра  
з теми: «РОЗРОБКА ІНТЕРФЕЙСУ ВЗАЄМОДІЇ КОРИСТУВАЧА ДЛЯ  
ПРОЕКТУ АВТОМАТИЗОВАНОГО РОБОЧОГО МІСЦЯ  
АДМІНІСТРАТОРА ГОТЕЛЮ»

Виконала: студентка 2 курсу, групи KN1-M18  
напряму підготовки 122 Комп'ютерні науки  
Бородай Тетяна Миколаївна

Керівник: Слободянюк О.В., старший викладач  
кафедри інформатики, кандидат технічних наук

Рецензент: Оптасюк С.В., доцент кафедри  
фізики, кандидат фізико-математичних наук

## ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1. ПОНЯТТЯ ІНТЕРФЕЙСУ ВЗАЄМОДІЇ КОРИСТУВАЧА .....	6
1.1. Особливості та типи взаємодії людини і комп'ютера.....	6
1.2. Поняття проектування взаємодії .....	15
1.3. Типи користувацького інтерфейсу .....	16
РОЗДІЛ 2. АНАЛІЗ ЗАСОБІВ ПРОЕКТУВАННЯ КОРИСТУВАЦЬКОГО ІНТЕРФЕЙСУ .....	23
2.1. Підходи до розробки User Interface .....	23
2.2. Бібліотеки та фреймворки для генерації інтерфейсу взаємодії .....	24
2.3. Порівняння інструментів для створення користувацького інтерфейсу .	27
РОЗДІЛ 3. РОЗРОБКА КОРИСТУВАЦЬКОГО ІНТЕРФЕЙСУ АВТОМАТИЗОВАНОГО РОБОЧОГО МІСЦЯ АДМІНІСТРАТОРА ГОТЕЛЮ .....	40
3.1. Етапи проектування .....	40
3.2. Розробка основних елементів користувацького інтерфейсу.....	41
ВИСНОВКИ .....	57
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	59

## ВСТУП

Скільки інтерактивних продуктів існує в повсякденному використанні. Задумайтеся на хвилину, що ви використовуєте кожного дня: смартфон, планшет, комп'ютер, пульт дистанційного керування, кавову машину, банкомат, квитковий автомат, принтер, iPod, GPS, TV, електричну зубну щітку, радіо, консоль ігор ... список можна продовжувати нескінченно.

Тепер подумайте на хвилину про те, наскільки вони корисні. Наскільки їх насправді легко і приємно використовувати? Деякі, як iPod, є радістю у використанні. Інші, як наприклад, квитковий автомат, може бути не дуже приємним у використанні. Різниця у тому, що такі пристрої як мобільний смартфон, зроблений для приємного використання людиною, щоб їй було в радість користування, тобто інтерфейс розроблений для користувача з урахуванням усіх вимог та бажань. А ось у квитковому автоматі має в собі незначний інтерфейс, тобто простий та зрозумілий, але він не зможе викликати ніяких емоцій, щоб було в задоволення використовувати.

З часу створення JavaScript суттєво змінилася розробка веб додатків із використанням цієї мови. JavaScript перестав бути допоміжним інструментом написання простих сценаріїв, а додатки сумішшю – HTML та JavaScript.

Створення великих і повноцінних JavaScript-додатків полягає в тому, що необхідно розділяти додаток на ряд практично незалежних компонентів. На кінцевий результат суттєво впливає вибір структури додатку, а саме шаблонний підхід. Шаблонний підхід дає змогу проектувати додатки випробуваним способом, що гарантує можливість ефективного обслуговування та розширення. Веб застосунки, що спроектовані без використання шаблонного підходу складніше супроводжуються, оскільки зміни в користувацькому інтерфейсі, логіці чи даних, можуть нести за собою зміни в інших частинах додатку, гірше тестуються та зменшується можливість повторного використання коду.

Для розробки веб додатків використовуються JavaScript фреймворки та застосовуються коли дуже важко чи довго виконувати поставлене завдання звичайними засобами розробки. Проте в більшості випадків дані фреймворки використовуються для написання односторінкових веб-додатків. Багато відомих JavaScript фреймворків, без яких важко уявити сучасну веб-розробку, використовують шаблони. Всі вони підтримують відповідні архітектурні шаблони, такі як MVC, MVP або MVVM.

**Актуальність** роботи полягає в процесі інформатизації роботи менеджерів у готельному бізнесі невеликого розміру.

**Метою** дипломної роботи є: розробка користувацького інтерфейсу автоматизованого робочого місця адміністратора готелю.

Для досягнення поставленої мети були сформульовані наступні завдання дослідження:

1) Розглянути поняття інтерфейсу користувача, надати основну класифікацію, та функціональні можливості для створення інтерфейсу користувача.

2) Дослідити користувацький досвід.

3) Описати технології та фреймворки для реалізації програмного застосунку.

4) Провести порівняльний аналіз найбільш популярних фреймворків.

5) Вибрати за допомогою якого фреймворка найкраще реалізовувати проект.

6) Реалізувати користувацький інтерфейс автоматизованого робочого місця адміністратора готелю за допомогою обраного веб фреймворку.

**Об'єктом** дослідження є процес розробки та проектування веб-додатку.

**Предметом** дослідження є методи розробки та проектування користувацьких інтерфейсів веб-ресурсів.

При написанні даної роботи використовувались такі **методи дослідження** як аналіз теоретичного матеріалу та наукової літератури з проблеми, метод порівняння для встановлення відмінностей між основними типами шаблонних

підходів проектування веб-сервісів, а також методи синтезу, формалізації та узагальнення при формуванні висновків.

**Практична цінність** роботи полягає у розробці програмного засобу для автоматизації робочого місця адміністратора готелю із використанням обраного веб фреймворку.

Структурно робота складається зі вступу, теоретичної частини, що містить 3 розділи, висновки, список літератури, додаток.

## ВИСНОВКИ

Отже, вибір структури, а саме шаблонного підходу, суттєво впливає на кінцевий результат додатку. Він дає змогу проектувати додатки випробуваним способом, що гарантує ефективне обслуговування та розширення. Завдяки шаблонному підходу веб-застосунки легше супроводжуються, краще тестуються та збільшуються можливість повторного використання коду. Патерни широко використовуються в різних методиках розробки програмного забезпечення, оскільки вони допомагають писати більш ефективний програмний код та використовувати напрацьовані методи, надають додатковий рівень абстракції, а також спрощують взаємодію між розробниками. Для розробки веб-додатків використовуються JavaScript фреймворки та застосовуються коли дуже важко чи довго виконувати поставлене завдання звичайними засобами розробки. JavaScript фреймворки мають чітку структуру і реалізуються з використанням архітектурних шаблонів, наприклад, таких як MVC, MVP, MVVM.

У розділі 2 здійснено аналіз засобів для реалізації користувацького інтерфейсу, були розглянуті такі фреймворки як AngularJS, Backbone.js, Ember.js, ReactJS та Vue.js, що використовують для побудови додатків архітектурні шаблони MVC, MVP та MVVM. На основі проаналізованих фреймворків та бібліотек для створення веб-додатку було обрано Vue.js, оскільки він має відносно невеликий розмір та не потребує залежностей. Також даний фреймворк виділяється невисоким порогом входження, оскільки потребує для початку принаймні знання мов HTML та JavaScript; має високу швидкодію, досить розвинуту спільноту розробників, документацію, що містить багато прикладів та за популярністю на основі Google Trends посідає провідне місце у пошукових запитах. Для нього систематично виходять оновлення та є фреймворком з відкритим програмним кодом.

Отже, під час виконання магістерської роботи було:

1. Проаналізовано теоретичний матеріал та наукова література з даної проблеми.
2. Розглянуто основні відомості про шаблонний підхід у проектуванні веб-додатків.
3. Описано основні види архітектурних шаблонів, що застосовуються у веб-програмуванні.
4. Проведено порівняльний аналіз для встановлення відмінностей найбільш популярних у даний момент шаблонних підходів.
5. Розроблено веб-додаток із застосуванням шаблонного підходу на основі одного з програмних каркасів (фреймворків), що його використовує.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. What is HCI (human-computer-interaction) [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/Breae9S>.
2. Gillian Crampton Smith. “The Hand that Rocks the Cradle” / Gillian Crampton Smith., 1995.
3. Brignull H. WHAT IS INTERACTION DESIGN? [Електронний ресурс] / Harry Brignull // 2019 – Режим доступу до ресурсу: <https://cutt.ly/Breaw94>.
4. Lövgren, J. & Stolterman, E., 2004. Thoughtful interaction design: a design perspective on information technology. Massachusetts: The MIT Press.
5. Winograd, T., 1996. Bringing Design to Software. ACM Press.
6. Nielsen & Norman. Intranet Design Annual / Nielsen & Norman., 2014. – 314 с.
7. GARRETT, J. J. The Elements of User Experience: User-centered design for the web and beyond / GARRETT, J. J., 2010. – (2-ге видання).
8. COOPER, A., REIMANN, R., CRONIN, D. and NOESSEL, C., 2014. – (The essentials of interaction design (4-те видання)).
9. Julia Carroll. Journal of Research in Reading [Електронний ресурс] / Julia Carroll – Режим доступу до ресурсу: <https://cutt.ly/CrearIq>.
10. McCarthy J. Technology as Experience / J. McCarthy, P. Wright., 2004.
11. Dewey J. The Collected Works of John Dewey [Електронний ресурс] / John Dewey – Режим доступу до ресурсу: <https://www.iep.utm.edu/dewey/>.
12. User Interface [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/xreaiLL>.
13. Аяx для новичков [Електронний ресурс]. – 2007. – Режим доступу до ресурсу: <https://cutt.ly/Areadq6>.
14. What is Interaction Design? [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/NreazU3>.



15. LIDWELL W. Universal Principles of Design / W. LIDWELL, K. HOLDEN, J. BUTLER., 2003.
16. NORMAN. The Design of Everyday Things: Revised and Expanded Edition / NORMAN., 2013.
17. SAFFER D. Designing for Interaction / SAFFER., 2010. – (2-ге видання).
18. Фреймворк или библиотека? А в чем, собственно, разница? [Электронный ресурс]. – 2019. – Режим доступа до ресурсу: <https://cutt.ly/4reaEIb>.
19. Отличие каркаса от библиотеки [Электронный ресурс]. – 2016. – Режим доступа до ресурсу: <https://cutt.ly/QreaURV>.
20. Програмный каркас [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/rreaTOA>.
21. И опять про MVC [Электронный ресурс]. – 2011. – Режим доступа до ресурсу: <https://cutt.ly/NreaSGX>.
22. Model-View-Presenter [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/mreaS81>.
23. Model-View-ViewModel [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/GreaGrR>.
24. Сравнение JavaScript фреймворков Vue.js, React и Angular [Электронный ресурс]. – 2019. – Режим доступа до ресурсу: <https://cutt.ly/vreaKyk>.
25. JavaScript libraries [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/5reaZSz>.
26. Knockout [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/PreaCNF>.

27. AngularJS [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/lresyHY>.
28. Backbone.js [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/qresuag>.
29. EmberJS [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/0resiyA>.
30. Kendo UI [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/jrespSa>.
31. Knockoutjs [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/1resa5M>.
32. ReactJS [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/gresfx4>.
33. Vue [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/Eresge4>.
34. Vue.js [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/bresxzo>.
35. Vue.js для сомневаючихся. Все, что нужно знать [Електронний ресурс]. – 2017. – Режим доступу до ресурсу: <https://cutt.ly/eresc88>.
36. How I stopped loving Angular [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://cutt.ly/RresRho>.
37. 10 советов и приемов, которые помогут вам стать лучшим разработчиком на VueJS [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/eresU3i>.
38. Веб-інтерфейс візуалізації геопросторових даних для мікросервісної платформи [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://cutt.ly/frekJ4m>.

39. Дослідження та аналіз особливостей реалізації компонентного підходу в сучасних Js фреймворках та бібліотек [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/8rekXPq>.

40. Установка Vue.js [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/UreQW0q>.

41. MLAB [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/dreEsDu>.

42. Фильтры Vue.js [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/hreRFoI>.

43. Добавление свойств экземпляра [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/JreRCOd>.

44. Фильтры Vue.js [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/hreRFoI>.

45. Добавление свойств экземпляра [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/JreRCOd>.

46. Form Builder with vue.js [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/VreTP7D>.

47. Аутентификация Vue и обработка маршрута с помощью Vue-router [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/0reTDAI>.

48. Vue.js аутентификация и обработка маршрутов с использованием Vue-router [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/UreTJrt>.

49. Bootstrap Vue [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/4reT0a7>.

50. Хостинг [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/BreYPTz>.

51. HeroKu [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/FreYSDh>.

52. Волков Ю.Ф. Технология гостиничного обслуживания. - Ростов-наДону: Феникс, 2003. - 384 с.

53. Node.js [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/5reUHqr>.

54. User Experience [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/8reUJ9d>.

55. Использование состояния Vuex в компонентах Vue [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/XreUCgQ>.

56. Хранение данных на стороне клиента [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/LreUVR8>.

57. Демонстрация повторно используемых компонентов React в формах [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/EreU25T>.

58. Framework comparison [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/greU5fW>.

59. AJAX [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/FreIrXB>.

60. Загрузка и запуск сайта на heroku.com [Электронный ресурс] – Режим доступа до ресурсу: <https://cutt.ly/LreIybl>.