

УДК [378.091.212:004]:[004.92:005.336.2]

Осадча Катерина Петрівна

кандидат педагогічних наук, доцент кафедри інформатики і кібернетики
Мелітопольський державний педагогічний університет імені Богдана Хмельницького,
м. Мелітополь, Україна
ORCID 0000-0003-0653-6423
okp@mdpu.org.ua

Чемерис Ганна Юріївна

аспірантка
Мелітопольський державний педагогічний університет імені Богдана Хмельницького,
м. Мелітополь, Україна
ORCID 0000-0003-3417-9910
Anyta.Chemeris@gmail.com

ФОРМУВАННЯ ГРАФІЧНОЇ КОМПЕТЕНТНОСТІ БАКАЛАВРІВ КОМП'ЮТЕРНИХ НАУК У ПРОЦЕСІ НАВЧАННЯ ПРОТОТИПУВАННЯ ПРОГРАМНИХ ІНТЕРФЕЙСІВ

Анотація. Стаття присвячена окремим аспектам процесу формування графічної компетентності у майбутніх бакалаврів комп'ютерних наук під час етапу прототипування програмного інтерфейсу в процесі розробки програмного забезпечення. Розглянуто сучасний стан та обґрунтовано актуальність етапу прототипування програмного інтерфейсу як посередника у людино-машинній взаємодії під час процесу розробки програмного забезпечення. Досліджено понятійний апарат у сфері прототипування, а саме поняття wireframe, prototure та mockup. Розглядаються чинники успішності проектів, проблеми у взаємодії учасників проекту і методи їх подолання. Наводиться порівняння сутності й аналіз основних методів прототипування і співвідношення їх вартості під час розробки програмного забезпечення. Проведено порівняльний аналіз програмного забезпечення й онлайн-ресурсів для прототипування за ступенем деталізації й з урахуванням базових характеристик засобу, типів та стадій прототипування, портативність і використання засобу. Визначено необхідний обсяг знань, умінь та навичок, що формуються під час вивчення таких дисциплін, як «Комп'ютерна графіка» і «Основи комп'ютерного дизайну», що є базовими під час вивчення прототипування програмних інтерфейсів. Встановлено їх зв'язок з етапами проектування програмного забезпечення. Наголошено на необхідності не лише у засвоєнні студентом обсягу теоретичних знань, а й успішного застосування у практичній діяльності. Результати дослідження можуть слугувати підґрунтям для удосконалення змісту навчально-методичного комплексу з дисципліни «Проектування програмного забезпечення» для бакалаврів комп'ютерних наук шляхом впровадження навчання прототипування від низькоточної до високоточної деталізації. Визначено вплив прототипування програмного інтерфейсу на формування професійних якостей майбутнього бакалавра комп'ютерних наук.

Ключові слова: прототипування; проектування; вайрфрейм; прототип% мокап; програмний інтерфейс; графічна компетентність; бакалаври з комп'ютерних наук.

1. ВСТУП

Постановка проблеми. В умовах сучасного інтенсивного розвитку ринку інформаційних технологій існує великий попит на висококваліфікованих фахівців у галузі комп'ютерних наук, здатних до адаптації в мінливому середовищі під час розв'язання завдань у різних сферах виробництва, підтримувати їх, пристосовуватись до конкретних практичних потреб, а також бути готовими приймати нові рішення і розробляти технології майбутнього. Під час процесу розробки програмного забезпечення чільне місце займає етап прототипування, як інструмент для уточнення

вимог замовника і швидкої розробки. Розробка прототипів уже давно є невід'ємною частиною у багатьох сферах суспільного життя, зокрема, у галузі архітектурного і виробничого проектування. Як зазначено О. Поморовою [1, с. 94], першим етапом під час процесу проектування будь-якого програмного забезпечення повинно бути створення прототипу, що передує програмуванню інтерфейсу програмного забезпечення. Прототипування є виключно цінним способом створення першої візуалізації проектів для демонстрації продукту, особливо на ранніх етапах тестування зручності застосування. Процес прототипування програмного забезпечення передбачає розв'язання низки задач, зокрема: зниження витрат на проектування, скорочення термінів проектування, покращення якості запропонованих рішень, забезпечення нескладного в освоєнні й використанні програмного забезпечення, вивчення та впровадження нових технологій і засобів, досягнення кращих результатів порівняно з конкурентами. Задоволеність користувача програмним продуктом або зручністю його використання значною мірою визначається інтерфейсом програмного забезпечення [1, с. 17].

Прототипування, як було зазначено В. Силантьєвим, полягає в створенні широкого набору прототипів майбутнього програмного інтерфейсу, які піддаються порівняльному аналізу й обговоренню. Мета прототипування полягає у визначенні, наскільки те чи інше рішення перспективне, з подальшою реалізацією одного з найкращих варіантів. Цей підхід дозволяє заощадити час і ресурси, що витрачаються на проектування і розробку [2, с. 146]. Однак у галузі розробки програмного забезпечення ця технологія почала використовуватися нещодавно, проте якнайкраще сприятиме формуванню навичок майбутньої професійної діяльності студентів. Слід припустити, що пов'язано це з тим, що в розробці програмного забезпечення акцент ставиться, у першу чергу, саме на технології і розробку функціональності, а не на проектуванні програмного інтерфейсу. Важливо зауважити, що процес прототипування не має єдиної технології для різних видів ресурсів. Однак існує низка апробованих на практиці методик і принципів, яким можна слідувати в процесі розробки програмного забезпечення [3].

Існує попит на здатних до прототипування фахівців у галузі комп'ютерних наук, які мають демонструвати інтеграцію технічних, інформаційних і дизайнерських навичок, уміти критично аналізувати великі обсяги інформації, приймати виважені рішення, працювати в команді, конструктивно реагувати на критику. Тому для реалізації навчання, в основі якого лежить ідея максимального забезпечення студентам їхніх шансів отримати гідне місце на ринку праці, підвищення їх особистісно-професійної цінності [4, с. 16], та спираючись на аналіз вітчизняних та закордонних стандартів підготовки бакалаврів комп'ютерних наук, що був здійснений у попередніх дослідженнях [11, с. 70-71] та, розглянувши визначені у освітніх програмах закладів вищої освіти до фахових компетентностей майбутніх бакалаврів з комп'ютерних наук, віднесено:

- здатність генерувати нові ідеї (креативність), що виявляється у знаннях основних етапів і стадій творчого процесу, ролі правильного формулювання мети і задач для їх досягнення у галузі комп'ютерних наук, творчі можливості людини, механізм генезису і розвитку знань, методи генерації ідей, розуміння креативності як універсального процесу породження незвичайних ідей;
- здатності проявляти допитливість, схильність до ризику, вміння мислити, надихатись новими ідеями, втілювати їх, запалювати ними оточуючих, комбінувати й експериментувати;
- уміння здійснення професійно-комунікативних контактів, розуміння співрозмовників, психологічний вплив в процесі комунікації, адекватне

розуміння вербальних і невербальних комунікативних сигналів, здатність долати комунікативні бар'єри.

Отже, можливо зробити висновок, що прототипування програмного інтерфейсу є важливим етапом у процесі проектування програмного забезпечення і професійній підготовці майбутніх бакалаврів з комп'ютерних наук.

Аналіз останніх досліджень і публікацій. Питання огляду сервісів прототипування знайшло своє відображення у роботах О. Проніної, Р. Баженова [5], А. Беляєва [6] та інших. Велику увагу процесу прототипування приділив Т. Варфел (T. Varfel) у своєму практичному керівництві з прототипування [7] та П. Хаст (P. Khast) у посібнику з проектування UX / UI [8]. Прототипуванню й розробці дизайну сайтів приділено увагу у дослідженнях М. Корзиної, В. Лисенко та А. Лисенко [9]. Загальну проблематику розробки дизайну сайтів було розглянуто І. Фроловим, Е. Горностаєвою [10], В. Силантьєвим [2] та у дослідженнях ефективності технологій прототипування в розробці веб-проектів Е. Притулою [3].

Мета статті. Дослідити використання прототипування програмного інтерфейсу у процесі формування графічної компетентності майбутніх бакалаврів комп'ютерних наук, дослідити закордонні і вітчизняні джерела з теми дослідження, проаналізувати понятійний апарат та існуючі офлайн і онлайн-засоби прототипування для подальшого розгляду під час вивчення дисципліни «Проектування програмного забезпечення».

2. МЕТОДИКА ДОСЛІДЖЕННЯ

З метою дослідження місця прототипування програмного інтерфейсу у процесі формування графічної компетентності бакалаврів комп'ютерних наук було досліджено понятійний апарат у галузі прототипування. З метою виявлення доцільних засобів прототипування було здійснено аналіз існуючого програмного забезпечення та онлайн-засобів прототипування. Результати проведення цього аналізу надали змогу визначити найбільш відповідний засіб прототипування програмного інтерфейсу для впровадження його розгляду в процесі професійного навчання майбутніх бакалаврів комп'ютерних наук під час вивчення дисципліни «Проектування програмного забезпечення» з метою формування графічної компетентності.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

3.1. Дослідження сучасного стану прототипування веб-проектів

Освіта майбутніх бакалаврів комп'ютерних наук повинна бути підпорядкована сучасним вимогам ринку праці сучасного постіндустріального інформаційного суспільства. Проте, незважаючи на розвиток галузі комп'ютерних наук і програмної інженерії, сьогодення демонструє нам динаміку, що виявляє високу частку провальних розроблюваних проектів [11, с. 72-75]. У практичному посібнику з прототипування Т. Варфел (T. Varfel), засновник і головний проектувальник взаємодії у компанії Messagefirst – консультативного агентства з проектування, стверджує, що на початку роботи з проектом єдине, що є у наявності — це безліч гіпотез, які треба перевіряти ще до того, як на розробку будуть витрачені значні кошти і час. Розв'язання цієї проблеми він однозначно вбачає у реалізації прототипу [8]. Серед усіх можливих причин відсутності комерційного успіху розроблюваних проектів найбільш суттєвими, за думкою А. Вендрова, є такі [12]: неконкретне і неповне формулювання вимог до програмного засобу; недостатня зацікавленість користувача у роботі над проектом;

відсутність необхідних ресурсів; незадовільне планування і безграмотне управління проектом; часта зміна вимог і специфікацій; новизна і недосконалість використовуваної технології; недостатня підтримка вищим керівництвом; недостатньо висока кваліфікація розробників, відсутність у них необхідного досвіду. Велику кількість із зазначених вище причин припинення розробки можливо уникнути на початковому етапі роботи над проектом під час етапу прототипування програмного інтерфейсу.

Нині існує безліч засобів прототипування, що розв'язують ці проблеми й покликані знижувати марні витрати часу і коштів, але низький відсоток успішності програмних проектів говорить про недостатність обґрунтування ефективності застосування цих засобів у розробці програмного забезпечення. Безсумнівно, прототипування — додатковий обсяг роботи, проте, це є технологією, що дозволяє заощадити як час, так і зменшити витрати. Прототип корисний тим, що надає можливість оцінити логіку роботи програмного забезпечення в цілому на будь-якому етапі реалізації проекту без прив'язки до дизайну.

Проте, за всього різноманіття існуючих засобів для створення прототипів, за даними, наведеними у дослідженні [3], у багатьох проектних організаціях воно досі не виконується. Більшість команд розробників програмного забезпечення мають добре налагоджений процес розробки й опрацьовують усі етапи створення програмного забезпечення до дрібниць – починаючи з технічного завдання і закінчуючи юридичною стороною. Але в більшості з них відсутній етап прототипування програмного інтерфейсу або виконується людьми, що не володіють необхідними знаннями в галузі технологій програмування (рис. 1).



Рис. 1. Наявність етапу прототипування і відповідальність

Мета прототипування полягає у визначенні, наскільки те чи інше рішення перспективне, і його подальша реалізація одна з найкращих. Цей підхід дозволяє заощадити час і ресурси, що витрачається на проектування і розробку [2]. З огляду на все вище зазначене, ми вважаємо, що вміння прототипувати є необхідною складовою графічної компетентності майбутнього бакалавра комп'ютерних наук, як інтегративної властивості, що включає вміння здійснювати якісну організацію і проектування програмних засобів, професійно використовувати сучасні комп'ютерні графічні технології під час проектування інтерфейсу, знаходити оптимальні технології досягнення кращого результату з урахуванням ергономічних вимог кінцевої аудиторії, тобто користувачів засобу чи споживачів продукту [12, с. 43]. Отже, вважаємо за

необхідне впровадження прототипування, як обов'язкового етапу в циклі професійної підготовки майбутніх бакалаврів комп'ютерних наук.

3.2. Огляд предметної галузі наукового дослідження

З метою визначення понятійного апарату, яким будемо оперувати у дослідженні, розглянемо детальніше термінологію. Прототипування полягає в створенні широкого набору макетів (прототипів) майбутнього програмного інтерфейсу, які піддаються аналізу. Як правило, прототип містить реалізацію лише самого інтерфейсу, без його функціонального наповнення [2].

Етап прототипування, на якому створюються структурні схеми сторінок (wireframes), що показують, яка інформація й елементи керування повинні розташовуватися на сторінках системи, є найважливішим у проектуванні інтерфейсу. Це ще не дизайн, але вже основа для нього, технічне завдання [14].

На сучасному етапі розвитку сфери розробки програмного забезпечення є проблема змішування термінів як синонімічних, а саме поняття “вайрфрейм”, “прототип” та “мокап”, визначаючи їх як виконаний із сіруватих блоків начерк конкретної ідеї. Оскільки відсутнє конкретне визначення зазначених термінів, наведемо власне тлумачення цих понять.

Вайрфрейм (Wireframe) – це образ дизайну низької точності (lo-fi), що візуалізує дизайн-проект гранично точно і допомагає отримати зворотній зв'язок на початковому етапі проектування програмного забезпечення. Вайрфрейми не лише набір сірих блоків, а й база майбутньої системи, що зображує кожну деталь фінального проекту. Він повинен чітко визначати основну групу контенту, структуру інформації й описувати базову візуалізацію взаємодії між інтерфейсом і користувачем. Вайрфрейм не містить занадто багато деталей, але створює точний образ фінального продукту, без втрати жодної важливої деталі, та є фундаментом майбутнього проекту і путівником для розробників, графічних дизайнерів, менеджерів проекту. Візуалізація повинна бути побудована за правилами естетики, але сильно спрощена завдяки схематичному відображенню.

Вайрфрейми використовуються у складі документації до проекту. Оскільки вони є статичним і фіксованим способом відображення інтерфейсу, тому вони потребують детального опису, наприклад як частина технічного завдання. Як зазначив Л. Буше (L. Busche), розробка вайрфреймів низької точності “допомагає знайти середину між перевитратами та переосмисленням, адже вони вимагають занадто мало інвестицій та у результаті надають занадто багато тестування користувачами” [15].

Прототип (Prototype) – це середньо-високе точне відображення (lo-hi) кінцевого продукту, що є динамічною симуляцією остаточної взаємодії між користувачем та інтерфейсом. Воно може не виглядати як кінцевий продукт, але має за мету надати користувачеві уявлення щодо вмісту і взаємодії з інтерфейсом, і надає можливість перевірити основний функціонал програмного забезпечення. Таке моделювання дій користувача формує основу для перевірки зручності використання інтерфейсу (usability-тестів) до початку розробки.

Взаємозалежність інтерфейсу і програмного забезпечення часто відсутня з метою скорочення витрат і пришвидшення циклу розробки. Прототипування - досить дорога за собівартістю й витратами людино/годин форма комунікації, тому доречно створення прототипів, що можуть бути використані в процесі розробки, тобто з використанням мови гіпертекстової розмітки HTML та каскадних таблиць стилів CSS.

Мокап (Mockup) – високо деталізований (hi-fi) статичний дизайн-образ інтерфейсу програмного засобу, що використовується для презентації загального відображення

дизайну кінцевого продукту. Як вайрфрейм, мокап статичний, але має багатші візуальні елементи. Він зосереджений на візуалізації продукту шляхом додавання багатьох візуальних елементів, щоб досягти вищої точності, ніж вайрфрейм, включаючи графіку, макет, колір, шрифтове вирішення та іншу більш детальну візуальну презентацію. Часто мокап - це вже актуальний графічний дизайн, що використовується в дискусіях з візуального представлення продукту для швидкого зворотного зв'язку та вдосконалення візуального дизайну програмного інтерфейсу.

Порівняння сутності та співвідношення методів прототипування наведено на рис 2.

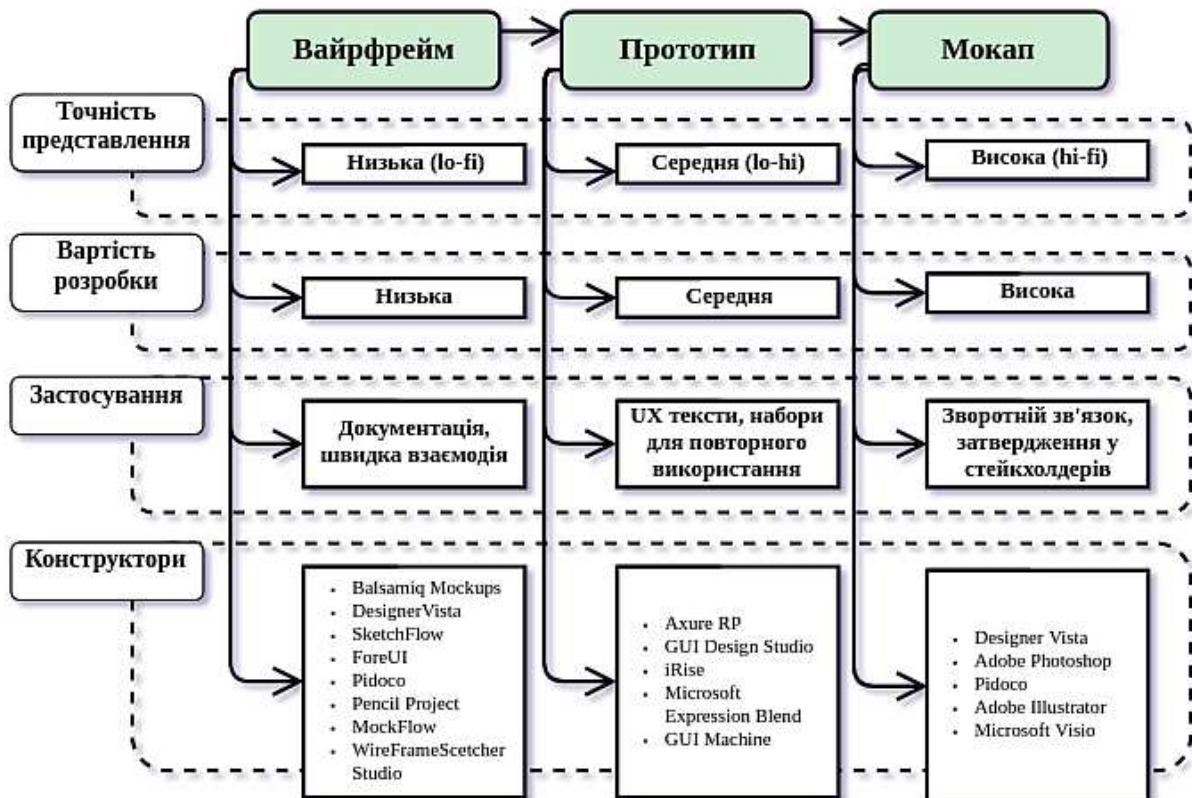


Рис. 2. Порівняння сутності понять вайрфрейм, прототип та мокап

Проте у своїй роботі конструктор взаємодії М. Реттіг (M. Rettig) вказує на деякий ризик у роботі з мокапами, адже є велика ймовірність критики вибору шрифтів, колірних комбінацій, розмірів кнопок тощо, нехтуючи формулюванням недоліків щодо концепцій. Замість того, щоб зосередити увагу на підтвердженні основних концепцій проекту, прототипи високої точності перенаправляють увагу на естетику продукту. З іншого боку, вайрфрейм "змушує користувачів думати про зміст, а не про зовнішність" [16]. Тож не можливо остаточно визначити який метод прототипування оптимальний, адже кожен з них має як свої недоліки, так і значні переваги.

Так, наприклад, на думку В. Силант'єва [2], прототипування інтерфейсів на папері в ХХІ столітті є найшвидшим, найдешевшим, кросплатформним та загальнодоступним методом. Науковець зазначає, що значна кількість розробників не віддають перевагу створенню паперових прототипів у результаті недовіри. Проте, виходячи з дослідження, що було проведено (M. Walker, L. Takayama, J. Landay) [17], прототипування програмними засобами, так і за допомогою олівця та паперу, дозволяє проводити тестування динамічно, і, як було у рамках цього дослідження

експериментально підтверджено, паперовий і комп'ютерний носії мають однакову силу для тестування прототипів.

Але не зважаючи на окремі переваги у використанні під час прототипування програмного інтерфейсу за допомогою олівця та паперу, застосування під час прототипування спеціалізованих програмних засобів та онлайн-ресурсів є, на наш погляд, більш доречним з огляду на можливість детально описати взаємодію користувача, продемонструвати інтерактивність, та працювати над проектуванням програмного засобу командою у режимі спільного доступу. Виходячи з усього вище зазначеного, розглянемо метод прототипування програмного інтерфейсу на папері у порівняльному аналізі поряд зі спеціалізованими засобами.

3.3. Аналіз інструментарію для прототипування програмного інтерфейсу

Розв'язання задачі формування графічної компетентності майбутнього бакалавра з комп'ютерних наук вимагає обґрунтованого вибору засобів. Оскільки головною проблемою навчання майбутнього бакалавра комп'ютерних наук є необхідність не лише у засвоєнні студентом обсягу теоретичних знань, а й успішного застосування цих знань у майбутній професійній діяльності, за доцільне вважаємо використання у процесі навчання актуальних засобів прототипування, що активно використовуються у виробництві.

Тому нами було здійснено пошук й аналіз наявних ресурсів для прототипування програмного інтерфейсу, результати якого наведено у таблиці 1. Ми ставили за мету знайти та проаналізувати як онлайн-ресурси, так і програмне забезпечення для персонального комп'ютера. Як критерій класифікації нами було визначено декілька груп. Так до основних характеристик ми долучили *мову* інтерфейсу; *обсяг* хмарного сховища у випадку онлайн доступу за умови безкоштовного використання; *кількість проектів*, що можливо зберігати у випадку онлайн доступу за умови безкоштовного використання, *платформа*, а саме наявність версій для ОС Windows, Linux, Mac, Web та мобільний додаток; *вартість* використання для 1 користувача на 1 рік, або на один місяць.

За умовні позначення нами було обрано: ∞ – безліч, ● - ідеально, ◐ - можливо застосувати, але не ідеально, ○ - неможливо, 🐧 - наявність версії для ОС сімейства Linux, 🍏 - наявність версії для ОС Mac, 🍷 - наявність версії для ОС Windows, 🌐 - наявність Web - версії, 📱 - наявність мобільного додатка або мобільної версії для опрацювання або перегляду прототипу.

Таблиця 1

Порівняльний аналіз програмних засобів та онлайн-ресурсів для прототипування

Деталізація	Lo				Lo - hi										Hi													
	Олівень	Mockingbird	Mockups	MockFlow	Proto.io	Axure	Balsamiq	Atomic	Flinto	Sketchapp	Veetr	Keynotoria	Mockplus	Uxpin	Justimind	Adobe Xd	Xcode	Webflow	Principle	Origami	Marvel	Justimind	Fluidui	Form	Gramer	Tumblhype	Invision	
Мова інтерфейсу	-	Англ	Англ	Англ	Англ	Англ	Англ Рос	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	Англ	
Обсяг, Гб	∞	∞	1	25	0,1	∞	∞	∞	∞	∞	∞	∞	∞	2	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	
К-ть проєктів	∞	1-∞	10	1-∞	5	∞	1-25	∞	∞	∞	∞	∞	∞	1	∞	∞	∞	∞	∞	∞	2-∞	∞	∞	∞	∞	∞	∞	
Платформа	-- -- -	🌐	🌐	🌐 🌐 🌐	🌐	🌐 🌐 -	🌐 🌐 -	🌐	🌐	🌐 🌐	🌐 🌐	🌐 🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	🌐	
Найдешевший тариф, \$	-	12/міс	13/міс	69	24	289	12/міс	19	20	99	-	49	14/міс	9/міс	19/міс	9/міс	-	16/міс	129	-	12	19	8	-	12	49,9	15	
Вільний доступ	●	●	◐	●	●	●	◐	●	◐	◐	◐	◐	◐	◐	◐	◐	◐	●	◐	●	●	◐	◐	●	◐	◐	◐	
<i>Типи та стадії прототипування</i>																												
На папері	●	●	●	●	●	◐	●	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	●	◐	◐	◐	◐	◐	◐	◐
Цифрова	◐	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Інтерактивна	◐	◐	◐	◐	●	◐	◐	●	◐	●	●	●	●	◐	◐	●	●	●	●	●	◐	●	●	●	●	◐	●	●
Швидка	●	●	●	●	◐	●	●	●	●	◐	◐	●	●	●	●	◐	◐	◐	◐	◐	●	●	●	◐	◐	◐	●	●
Рання стадія	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	◐	◐	◐	◐	●	◐	●	●	◐	◐	●	●
Пізня стадія	◐	◐	◐	◐	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
<i>Портативність та використання</i>																												
Доопрацювання	●	●	●	●	●	◐	●	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐
Кодування	◐	◐	◐	◐	◐	●	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	●	●	●	●	◐	●	●	◐	●	●	◐	◐
Спільний доступ	◐	●	●	◐	◐	◐	●	●	◐	◐	●	◐	●	●	●	●	◐	◐	◐	◐	●	●	●	◐	◐	◐	●	●

Група характеристик за типом і стадіями прототипування: *на папері* — можливість представлення розробленого прототипу на папері, у вигляді роздруківки тощо, з подальшим оповідальним описом; *цифрова* — можливість представлення розробленого прототипу на цифрових носіях, а саме у вигляді статичних зображень; *інтерактивна* - представлення прототипу на цифрових носіях, а саме у вигляді документів з гіперпосиланнями, інтерактивних проєктів, відеопрезентація тощо, та здатність прототипу реагувати на дії користувача і емулювати реальні події; *швидка* — швидкість створення прототипу, можливість реалізувати концепцію інтерфейсу без знання технологій; *рання стадія* — доречність використання на початку роботи з проєктом, в особливості здатність до швидкого представлення базової концепції без детального опрацювання; *пізня стадія* — доречність використання під час остаточної роботи над програмним інтерфейсом, з детальним опрацюванням усіх дрібниць графічного відображення, взаємодії з користувачем та навіть з опрацюванням фрагментів коду.

Група характеристик за *портативністю та використанням: доопрацювання* — повторне опрацювання прототипу з метою деталізації та конкретизації, опрацювання дрібниць графічного відображення, взаємодії з користувачем та опрацювання фрагментів коду; *кодування* — наявність можливості формування й опрацювання окремих фрагментів коду; *спільний доступ* — доступність для всіх учасників проєкту можливість коментування замовником, керівництвом для прояснення окремих деталей майбутньої системи, та можливість спільного редагування розробниками. Прототип повинен оновлюватися разом з проєктними артефактами, тому можливість внесення змін - важливий критерій.

Майже усі засоби надають можливість ознайомитись з функціоналом під час демонстраційного періоду, під час якого надається можливість роботи з програмним засобом з обмеженим функціоналом або з обмеженням строку використання. Проте є програмне забезпечення з повністю вільним доступом, а саме це Origami, Form, Vectr, Xcode, Atomic, та з умовно вільним доступом, як Moqups, обмеження використання якого на безоплатній основі накладаються на обсяг використовуваних об'єктів та кількість проєктів. Також універсальним і гнучким засобом, за допомогою якого можливо реалізувати як низькоточне представлення так і високоточне є Sketchapp.

З огляду на усе вищезазначене, робимо висновок, що найкращим засобом для прототипування програмного інтерфейсу для роботи в аудиторному режимі у процесі навчання майбутніх бакалаврів комп'ютерних наук, на наш погляд, є Balsamiq, Moqups, Vectr та Webflow.

3.4. Прототипування у дисциплінах професійного циклу навчання

Згідно з проведеним аналізом, результати якого наведені у таблиці 1, існує чимало різних інструментів і методів для створення прототипів програмного інтерфейсу на різних етапах розробки, проте низький відсоток завершених розробок говорить про недостатність вивчення технології прототипування й обґрунтування ефективності її застосування.

У даний час представлено чимало друкованих та Інтернет-джерел, присвячених розробці програмного забезпечення та прототипуванню для промисловості, але практично немає джерел, що описують процес створення і використання прототипів, для фахівців, що займаються розробкою програмного забезпечення. До найважливіших професійних якостей майбутнього бакалавра з комп'ютерних наук належать вміння створити естетичний програмний продукт, користуючись для цього широким арсеналом засобів і прийомів.

У практичній діяльності він має розв'язувати творчі завдання і демонструвати здібності графічної компетентності, розробляючи прототипи користувацького інтерфейсу, за допомогою комп'ютерних програмних засобів втілювати задум у прототип майбутнього проекту, і обравши метод прототипування й підібравши усі необхідні параметри, втілити задум у якісному програмному засобі.

Одним із способів формування і розвитку компетентностей є навчання, побудоване на активній навчальній діяльності і проектно-орієнтованій діяльності студентів, що інтегрує фундаментальні знання і практичні вміння. Розглядаючи необхідність впровадження прототипування під час вивчення дисципліни "Проектування програмного забезпечення" майбутніми бакалаврами комп'ютерних наук необхідним вбачаємо важливість прототипування з низькою точністю, з поступовим переходом до безпосереднього інтерактивного та високодеталізованого прототипування (рис. 3).

Такий поступовий перехід від прототипування з низькою точністю до високої точності реалізує принцип гнучкої методології розробки проектних завдань у навчальному процесі (*Agile-методологія*) як серію ітерацій (циклів робіт), де кожна ітерація має вигляд програмного проекту в мініатюрі та включає всі завдання, необхідні для отримання приросту функціональності.

Процес деталізації прототипу програмного інтерфейсу від низькорівневого вайрфрейму до високорівневого мокапу під час опрацювання практичної роботи з дисципліни "Проектування програмного забезпечення" продемонструємо на прикладі програмного інтерфейсу веб-ресурсу, що надане П. Хастом (P. Khast) (Рис. 3,4) [9, с. 26].

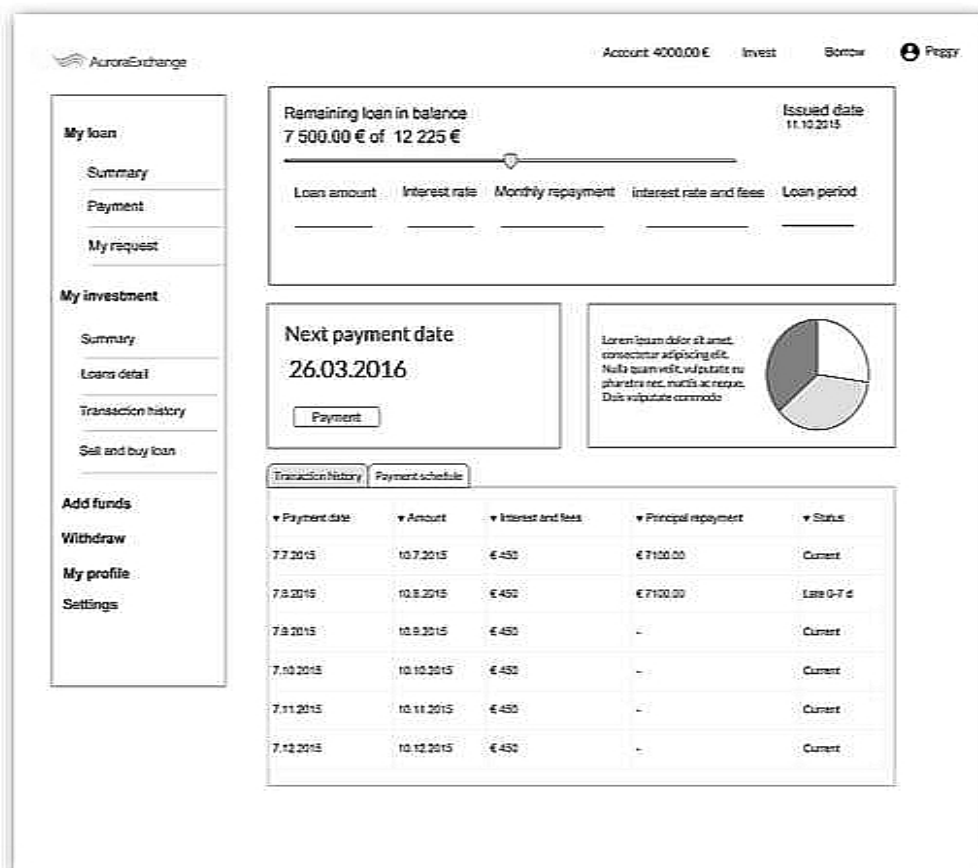


Рис. 3. Початковий етап опрацювання програмного інтерфейсу. Вайрфрейм

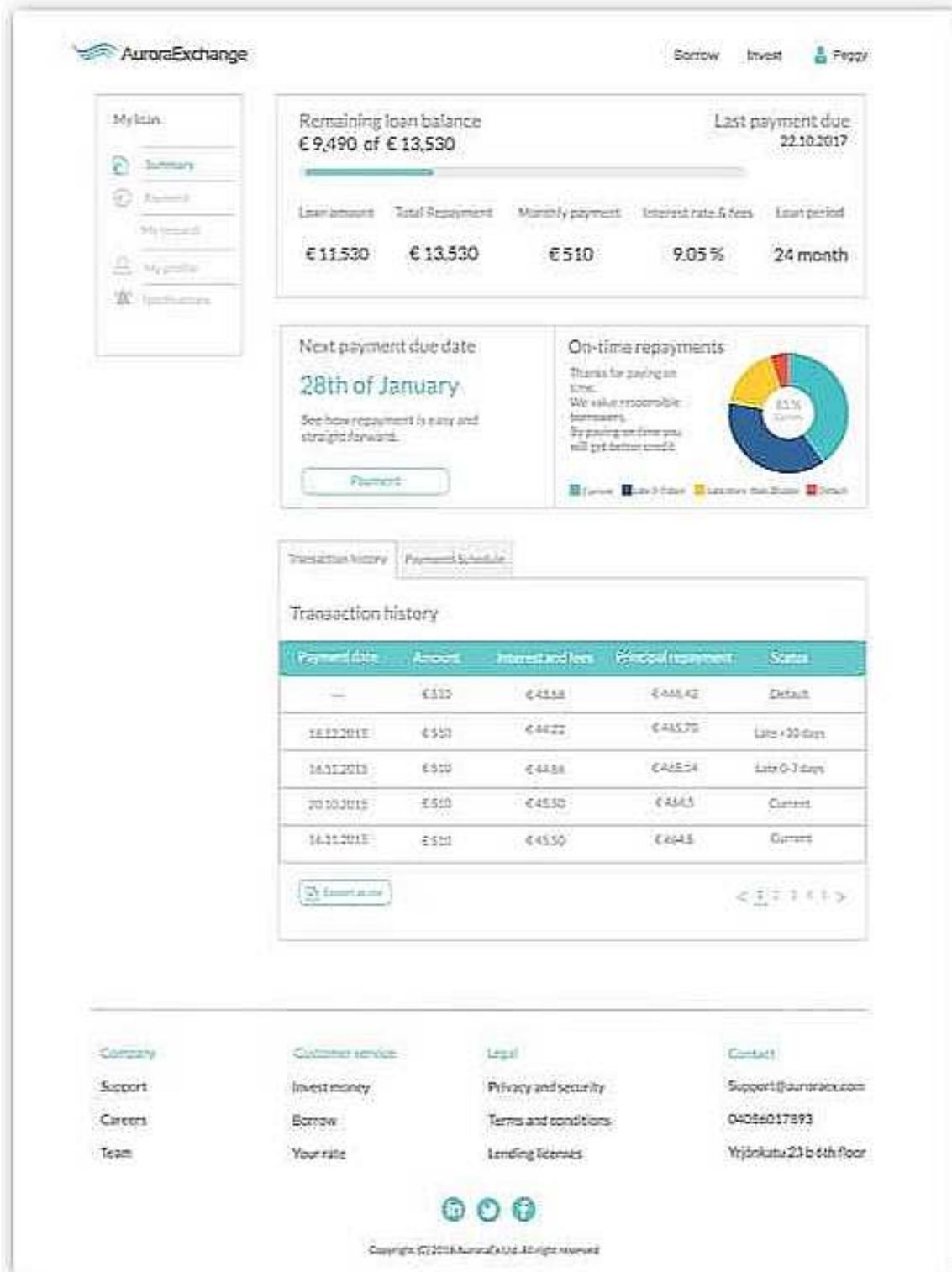


Рис. 4. Завершальний етап опрацювання програмного інтерфейсу. Мокап

Основою для такого еволюційного прототипування програмного інтерфейсу під час вивчення дисципліни “Проектування програмного забезпечення” є знання, вміння та навички, що формуються під час вивчення таких дисциплін, як “Комп’ютерна графіка”, “Основи комп’ютерного дизайну” та “Обробка зображень та мультимедіа”,

оскільки у процесі їх вивчення зустрічаються багато спільних понять, що уточнюються, конкретизуються, систематизуються, а саме:

- знання щодо *розуміння основ композиції* (розкриття ідейно-художнього змісту; розміщення основних елементів у визначеній системі та послідовності; способи поєднання образів у єдине ціле) для грамотного і гармонійного розташування змістових блоків програмного інтерфейсу;
- *знання базових понять та основ ергономіки* (взаємодія людина-система; принципи ефективної організації простору системи та її керування) для створення зручного у використанні програмного інтерфейсу й обґрунтованого вибору взаємодії;
- *знання закономірностей кольорознавства* (фізична теорія кольору; основні колірні системи; сприйняття кольору) для гармонійного вживання кольорових схем і колірних поєднань;
- *основи типографіки* (основи шрифтових поєднань; оптимізація типографіки) для доречного й обґрунтованого використання шрифтів та їх поєднань;
- *основи естетики* тощо.

Ці знання формують необхідну базу для подальшого вивчення графічних дисциплін й складають початковий рівень графічної компетентності майбутнього бакалавра з комп'ютерних наук. Вплив і зв'язок перерахованих навичок графічної компетентності продемонстровано (рис 5.).



Рис. 5. Зв'язок знань, умінь та навичок графічної компетентності з етапами проектування програмного забезпечення

Так під час роботи над прототипуванням студенти зможуть досліджувати альтернативи і, насамперед, зосередитись на основних аспектах і потребах цільової аудиторії з метою покращення взаємодії користувача з програмним інтерфейсом. Для більшої результативності у роботі над прототипуванням і розробкою програмного інтерфейсу студенти мають опрацювати не лише завдання з розробки інтерфейсу, а й провести аналіз, дослідити контент майбутнього проекту для визначення його структури, накопичити й проаналізувати пропозиції після тестування.

Організація практичних занять з дисципліни “Проектування програмного забезпечення” буде цікавою, мотивуючою і результативною, орієнтованою на майбутню професійну діяльність, якщо надавати студентам не просто завдання з розробки готових рішень програмного інтерфейсу, а націлювати на розв’язання ситуацій, наближених до реальних, а саме:

- взаємодія із замовником і робота в команді під час обговорення прототипу користувацького інтерфейсу у системі комунікації засобу прототипування (чат, коментування, спільний доступ та редагування тощо);
- розв’язання конкретних проблем прототипування користувацького інтерфейсу;
- ефективне і систематичне перетворення знань, що були зібрані впродовж досліджень, у детальну специфікацію інтерфейсу;
- знаходження й усунення невідповідностей на рівні прототипування програмного інтерфейсу, що дозволять глибоко і детально дослідити конкретну ситуацію, яка може виникнути у майбутній професійній діяльності для її уникнення на етапі програмування інтерфейсу тощо.

Результатом вивчення теми прототипування користувацького інтерфейсу у межах дисципліни “Проектування програмного забезпечення” є формування таких професійних якостей, як уміння застосовувати основи композиції, кольорознавства та ергономіки під час прототипування; розуміння основних концепцій, принципів та методів графічного подання й візуалізації програмного інтерфейсу; здатність до стилізації та формалізації з урахуванням контенту програмного засобу; формуванню у студентів основних положень теорії людино-машинної взаємодії в процесі розробки програмного забезпечення; практичні навички створювати програмні інтерфейси; формування навичок застосування засобів розробки інтерфейсу та навичок реалізації основних стадій процесу прототипування програмного інтерфейсу. Застосування базових навичок з прототипування програмного інтерфейсу майбутніми бакалаврами з комп’ютерних наук стане також у нагоді у процесі виконання наукових робіт під час розробки програмного забезпечення.

Необхідність вивчення і застосування технологій прототипування для успішної реалізації програмного інтерфейсу підтверджує не тільки актуальність теми дослідження, а й практичну цінність. Особистий досвід підтверджує ефективність впровадження прототипування програмного інтерфейсу в підготовці майбутніх бакалаврів комп’ютерних наук, особливо у роботі студентів під час розробки програмного забезпечення, оскільки передбачає розгляд значної кількості досвіду користувачів, реальних ситуацій, завдань в певних комбінаціях, та комунікації Виконавця із Замовником протягом усього процесу розробки проекту.

4. ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У процесі дослідження було розглянуто використання прототипування програмного інтерфейсу у процесі формування графічної компетентності майбутніх бакалаврів комп’ютерних наук. Визначено сутність термінів “вайрфрейм”, “прототип” та “мокап”. Були розглянуті чинники комерційного успіху проектів, проблеми у взаємодії учасників проекту та методи їх подолання. Наведено порівняльний аналіз основних методів прототипування і співвідношення їх деталізації та вартості під час розробки програмного засобу. Проаналізовано існуючі програмні засоби й онлайн-ресурси прототипування для подальшого розгляду під час вивчення дисципліни “Проектування програмного забезпечення”.

Результати дослідження можуть слугувати підґрунтям для удосконалення змісту навчально-методичного комплексу з дисципліни “Проектування програмного забезпечення” та встановлення міждисциплінарних зв’язків з дисциплінами “Комп’ютерна графіка”, “Основи комп’ютерного дизайну” та “Обробка зображень та мультимедіа”, зорієнтованого на впровадження практичних робіт з прототипування програмного інтерфейсу, що сприятиме формуванню графічної компетентності майбутніх бакалаврів комп’ютерних наук.

Перспективи подальших досліджень вбачаються у розробці навчально-методичних рекомендацій й удосконаленню змісту навчально-методичного комплексу з дисципліни “Проектування програмного забезпечення” щодо впровадження вивчення етапів прототипування програмного інтерфейсу майбутніми бакалаврами з комп’ютерних наук.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] О. В. Поморова и Т. О. Говорущенко *Проектування інтерфейсів користувача: навч. посібник*, Хмельницький, Україна: ХНУ, 2011.
- [2] В. Е. Силантьев "Этапы проектирования пользовательского интерфейса" *Системы обработки информации*, Вып. 9, с. 143-148, 2014.
- [3] Е. В. Прутула "Исследование эффективности технологии прототипирования в разработке web-проектов" *Прикладные информационные системы*, No 1, с. 32-40, 2014.
- [4] Ю. М. Рашкевич *Болонський процес та нова парадигма вищої освіти: монографія*. Львів, Україна: Видавництво Львівської політехніки, с. 168, 2014.
- [5] О. Ю. Проніна та Р. І. Баженов "Обзор сервисов по созданию прототипов" *Постулат*, No 12, 2016.
- [6] А. А. Беляев "Анимация в дизайне интерфейса информационных сайтов" *Медиаскоп*, No 4, с. 11, 2014.
- [7] T. Z. Warfel *Prototyping A Practitioner's Guide*, Rozenfeld Media, Brooklyn, NY, 2009.
- [8] P. Khast *UX/UI Design Process for a Peer to Peer Financial Platform*, 2017.
- [9] М. И. Корзина, В. А. Лысенко и А. А. Лысенко "Системное проектирование (дизайн) web-сайта" *Дизайн. Материалы. Технология*, No 1(26), с. 116-122, 2013.
- [10] Е. А. Горностаева "Современные проблемы дизайна сайтов и вызовы нового времени" *Молодой ученый*, No 1 (81), с. 38-40, 2015.
- [11] К. П. Осадча та Г. Ю. Чемерис "Добір засобів тривимірного моделювання для формування графічної компетентності майбутніх бакалаврів комп’ютерних наук" *Інформаційні технології і засоби навчання*, Том 62, No 6, 2017. [Електронний ресурс]. Доступно: <https://journal.iitta.gov.ua/index.php/itlt/article/view/1713/1273> Дата звернення: Березень 30, 2018.
- [12] К. П. Осадча та Г. Ю. Чемерис "Аналіз сутності поняття «графічна компетентність» у системі підготовки майбутнього бакалавра з комп’ютерних наук". *Ukrainian Journal of Educational Studies and Information Technology*, Vol. 5, №. 3, 2017. [Електронний ресурс]. Доступно: <http://www.ojs.mdpu.org.ua/index.php/itse/article/view/1929/2446> Дата звернення: Март 30, 2018.
- [13] А. М. Вендров "Современные технологии создания программного обеспечения". *Citforum* [Електронний ресурс]. Доступно: <http://citforum.ru/programming/application/program/index.shtml>, 2004. Дата звернення: Березень 18, 2018.
- [14] М. Б. Максимович "Технології проектування інтерфейсу користувача навчальних комп’ютерних систем" *Альманах*. Херсон, Україна: Видавництво ХДУ, Вип. 8, с. 84-86, 2011.
- [15] L. Busche "The Skeptic's Guide to Low-Fidelity Prototyping" *Smashing magazine*, 2014.
- [16] M. Rettig "Prototyping for tiny fingers" *Communications of the ACM*, T. 37, No. 4, с. 21-27, 1994.
- [17] M. Walker, L. Takayama & Landay J. A. "High-fidelity or low-fidelity, paper or computer? Choosing attributes when testing web prototypes" *Proceedings of the human factors and ergonomics society annual meeting*, SAGE Publications: Los Angeles, T. 46, No. 5, с. 661-665, 2002.

Матеріал надійшов до редакції 23.04.2018 р.

ФОРМИРОВАНИЕ ГРАФИЧЕСКОЙ КОМПЕТЕНТНОСТИ БАКАЛАВРА КОМПЬЮТЕРНЫХ НАУК В ПРОЦЕССЕ ИЗУЧЕНИЯ ПРОТОТИПИРОВАНИЯ ПРОГРАММНЫХ ИНТЕРФЕЙСОВ

Осадчая Екатерина Петровна

кандидат педагогических наук, доцент кафедры информатики и кибернетики
Мелитопольский государственный педагогический университет имени Богдана Хмельницкого,
г. Мелитополь, Украина
ORCID 0000-0003-0653-6423
okp@mdp.u.org.ua

Чемерис Анна Юрьевна

аспирант
Мелитопольский государственный педагогический университет имени Богдана Хмельницкого,
г. Мелитополь, Украина
ORCID 0000-0003-3417-9910
Anyta.Chemeris@gmail.com

Аннотация. Статья посвящена отдельным аспектам процесса формирования графической компетентности у будущих бакалавров компьютерных наук во время этапа прототипирования программного интерфейса в процессе разработки программного обеспечения. Рассмотрены современное состояние и обоснована актуальность этапа прототипирования программного интерфейса как посредника в человеко-машинной взаимодействия в процессе разработки программного обеспечения. Исследована понятийный аппарат в сфере прототипирования, а само понятие wireframe, prototype и mockup. Рассматриваются факторы успешности проектов, проблемы во взаимодействии участников проекта и методы их преодоления. Приводится сравнение сущности и анализ основных методов прототипирования и соотношение их стоимости при разработке программного обеспечения. Проведен сравнительный анализ программного обеспечения и онлайн - ресурсов для прототипирования по степени детализации и с учетом базовых характеристик средства, типов и стадий прототипирования, портативность и использования средства. Определен необходимый объем знаний, умений и навыков, которые формируются при изучении таких дисциплин, как "Компьютерная графика" и "Основы компьютерного дизайна", которые являются базовыми при изучении прототипирования программных интерфейсов. Установлено их связь с этапами проектирования программного обеспечения. Отмечена необходимость не только в усвоении студентом объема теоретических знаний, но и успешного применения в практической деятельности. Результаты исследования могут служить основой для совершенствования содержания учебно-методического комплекса по дисциплине "Проектирование программного обеспечения" для бакалавров компьютерных наук путем внедрения обучения прототипирования от низкоточных к высокоточной детализации. Определено влияние прототипирования программного интерфейса на формирование профессиональных качеств будущего бакалавра компьютерных наук.

Ключевые слова: прототипирование; проектирование; вайрфрейм; прототип; мокап; программный интерфейс; графическая компетентность; бакалавры компьютерных наук.

FORMATION OF COMPUTER SCIENCE BACHELOR'S GRAPHIC COMPETENCY WHILE STUDYING THE PROTOTIFICATION OF SOFT INTERFACES

Kateryna P. Osadcha

PhD of Pedagogical Sciences, Associate Professor of the Department of Informatics and Cybernetics
Bogdan Khmelnytsky Melitopol State Pedagogical University, Melitopol, Ukraine
ORCID 0000-0003-0653-6423
okp@mdp.u.org.ua

Hanna Yu. Chemerys

PhD student

Bogdan Khmelnytsky Melitopol State Pedagogical University, Melitopol, Ukraine

ORCID 0000-0003-3417-9910

Anyta.Chemeris@gmail.com

Abstract. The article deals with some aspects of the process of forming the future computer science bachelor's graphic competency at the stage of user's interface prototyping in the course of software development. The modern situation has been studied and the urgency of the stage of user's interface prototyping as a mediator in the man-machine coordination during the process of software development is proved. The Conceptual set in the sphere of prototyping is examined, namely — the concepts of wireframe, prototype and mockup. The factors of projects successfulness, the problems in the coordination of the project participants and the methods of their overcoming are being considered. The comparison of the essence and the analysis and the main methods of prototyping and correlating their value during the software development are given. The comparative analysis of the software and on-line resources for prototyping according to the detalization degree and taking into account the basic characteristics of a means, types and stages of prototyping, portability and using the means is carried out. The necessary base of knowledge, skills and knacks, forming during the study of such disciplines as “Computer Graphics” and “The Essential of Computer Design”, as well as essentials of composition, colour theory, ergonomics and typography is determined. The necessity of both the student's mastering an amount of theoretical knowledge and its successful using in his practical activity is stressed. The results of the research can be a substratum for improving the content of the educational and methodical complex in the discipline “Software Projecting” for computer science bachelors by means of introducing the teaching of prototyping from the low-fidelity to high fidelity detalization. The influence of user's interface prototyping on the formation of the future computer science bachelor's professional qualities is determined.

Keywords: prototyping; designing; wireframe; prototype; mockup; user interface; graphic competency; bachelors in computer sciences.

REFERENCES (TRANSLATED AND TRANSLITERATED)

- [1] O. V. Pomorova y T. O. Hovorushchenko *Design of user interfaces: curriculum. manual*, Khmelnytskyi, Ukraina: KhNU, 2011. (in Ukrainian)
- [2] V. E. Silant'ev "Stages of User Interface Design" *Systemy obrobky informatsii*, Вип. 9, p. 143-148, 2014. (in Ukrainian)
- [3] E. V. Pritula “Study of the Efficiency of Prototyping Technology in the Development of Web Projects” *Prikladnye informacionnye sistemy*, No 1, p. 32-40, 2014.(in Russian)
- [4] Iu. M. Rashkevych *Bologna Process and New Paradigm of Higher Education: Monograph*. Lviv, Ukraina: Vydavnytstvo Lvivskoi politekhniky, p. 168, 2014. (in Ukrainian)
- [5] O. Ju. Pronina ta R. I. Bazhenov "An overview of services for the creation of prototypes" *Postulat*, No 12, 2016. (in Russian)
- [6] A. A. Beljaev "Animation in the design of the interface of information sites" *Mediaskop*, No 4, p. 11, 2014. (in Russian)
- [7] T. Z. Warfel *Prototyping A Practitioner's Guide*, Rozenfeld Media, Brooklyn, NY, 2009. (in English)
- [8] P. Khast *UX/UI Design Process for a Peer to Peer Financial Platform*, 2017. (in English)
- [9] M. I. Korzina, V. A. Lysenko & A. A. Lysenko "System design of the web-site" *Dizajn. Materialy. Tehnologija*, No 1(26), p. 116-122, 2013. (in Russian)
- [10] E. A. Gornostaeva "Modern Problems of Website Designing and Challenges of the New Time" *Molodoj uchenyj*, No 1 (81), c. 38-40, 2015. (in Russian)
- [11] K. P. Osadcha ta H. Yu. Chemerys "Three-dimensional modeling tools in the process of formation of graphic competence of the future bachelor of computer science" *Information Technologies And Learning Tools*, Vol 62, No 6, p. 70-85, 2017. [Online]. Available: <https://journal.iitta.gov.ua/index.php/itlt/article/view/1713/1273> Accessed on: March 30, 2018. (in Ukrainian)
- [12] K. P. Osadcha ta H. Yu. Chemerys "Analysis Of The Mention Of The Concept “Graphic Competence” In The Preparation Of The Future Bachelor Of Computer Science”. *Ukrainian Journal of Educational Studies and Information Technology*, Vol. 5, №. 3, p. 37-46, 2017. [Online]. Available:

- <http://www.ojs.mdpu.org.ua/index.php/itse/article/view/1929/2446> Accessed on: March 30, 2018. (in Ukrainian)
- [13] A. M. Vendrov "Modern Software Development Technologies". *Citforum* [Online]. Available: <http://citforum.ru/programming/application/program/index.shtml>, 2004. Accessed on: March 18, 2018. (in Russian)
- [14] M. B. Maksymovych "Technologies of Designing the User Interface for Learning Computer Systems" *Almanakh*. Kherson, Ukraina: Vydavnytstvo KhDU, Вип. 8, с. 84-86, 2011. (in Ukrainian)
- [15] L. Busche "The Skeptic's Guide to Low-Fidelity Prototyping" *Smashing magazine*, 2014. (in English)
- [16] M. Rettig "Prototyping for tiny fingers" *Communications of the ACM*, Т. 37, No . 4, с. 21-27, 1994.(in English)
- [17] M. Walker, L. Takayama & Landay J. A. "High-fidelity or low-fidelity, paper or computer? Choosing attributes when testing web prototypes" *Proceedings of the human factors and ergonomics society annual meeting*, SAGE Publications: Los Angeles, Т. 46, No . 5, с. 661-665, 2002. (in English)



This work is licensed under Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.