

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

# **ПРОТОТИПУВАННЯ КОРИСТУВАЦЬКИХ ІНТЕРФЕЙСІВ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

## **МЕТОДИЧНІ ВКАЗІВКИ**

до лабораторної роботи № 2  
із курсу «Людино-машинна взаємодія»  
для студентів бакалаврів 6.122 – Комп'ютерні науки

*Затверджено  
на засіданні кафедри  
Системи штучного інтелекту.  
Протокол № 1 від \_\_\_\_\_*

**Львів 2019**

**Прототипування користувацьких інтерфейсів програмного забезпечення.** Методичні вказівки до лабораторної роботи № 2 з курсу «Людино-машинна взаємодія» для студентів бакалаврів напрямку 6.122 – Комп'ютерні науки / Укл. Маркелов Олександр Едуардович – Львів: Видавництво Національного університету “Львівська політехніка”, 2019. – 20 с.

**Укладач:** Маркелов Олександр Едуардович, доцент, к.т.н.

**Відповідальний за випуск:**

**Рецензенти:**

## 1. МЕТА ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

*Метою виконання лабораторної роботи* є ознайомлення зі способами, методами та засобами прототипування користувацьких інтерфейсів програмного забезпечення та набуття студентами практичних навичок автоматизованого створення сценаріїв інтерактивної взаємодії користувача, створення різного виду прототипів функціонування діалогів інтерфейсів із наборів конструкторських елементів інтерфейсу користувача.

У результаті виконання лабораторної роботи студенти повинні:

- *знати* методи та підходи для створення прототипів користувацьких інтерфейсів, їхні властивості, параметри, варіанти застосування та реалізації;
- *вміти* правильно задавати значення властивостей елементів для різних типів інтерактивної взаємодії; правильно вибирати керуючі події елементів.

## 2. ТЕОРЕТИЧНІ ВІДОМОСТІ

### Основні поняття

В англійській термінології для позначення понять ескізного проектування інтерфейсів користувача часто використовують слова: **Prototype, Mock-Ups, Wireframe, Sketching**. А для сценаріїв взаємодії: **Use Case, Usage Scenarios**.

**Прототип** проекту інтерфейсу – це версія програми або веб-сайту у спрощеному вигляді. Прототипи, створювані в спеціалізованих редакторах, зосереджуються на рівні користувацького інтерфейсу (UI). Вони містять особливості графічного інтерфейсу користувача, а також принципи та схеми навігації, але не логіку чи структуру бази даних, програмний код логіки внутрішньої функціональності. Часто такі прототипи називаються каркасними прототипами або просто каркаси, коли вони абстрагуються від деталей графічного дизайну і замість цього зосередитися на планування і структуруванні графічного інтерфейсу користувача, з використанням блоків-наповнення, які виглядають як прямокутники. Прототипи можуть бути використані для візуалізації та передачі ідей або концепцій і для проведення тестування зручності.

Прототип дозволяє показати розміщення елементів інтерфейсу на екрані пристрою. Описує кожний елемент та його поведінку, відгук на дії користувача. Основна увага в прототипі фокусується саме на розміщенні, написах, взаємодіях та переходах. **Прототипування уникає кінцевого графічного оформлення, як кольори та зображення, щоб не зосереджувати увагу на красі, а більше акцентуватися на виконуваних функціях,**

**призначенні. Кінцеве графічне дизайнерське оформлення це наступна стадія розробки інтерфейсу користувача.**

Прототип повинен у першу чергу відображати функціональність інтерфейсу результуючої системи, тому його перші версії роблять досить «примітивними».

### **Принципи прототипування діалогових систем користувача**

Прототипи є засобом для генерації ідей про те, як можна реалізувати інтерфейси користувача. І це допомагає оцінити якість рішення на ранній стадії. Є кілька підходів до прототипування. Важливо визначити мету будь-якого прототипу при побудові (таблиця 2.1). Коли здійснюються покращення або зміни існуючої системи, то еволюційні прототипи дозволяють відчувати, як наступна версія програмного продукту буде виглядати. Для проектування користувацького інтерфейсу з нуля є експериментальні та дослідні прототипи з відповідними засобами для проведення процесу проектування. [2]

*Таблиця 2.1*

#### **Варіанти мети макетування інтерфейсів**

Мета	Опис
Еволюційне макетування	<ul style="list-style-type: none"><li>• Постійно адаптувати систему до швидкої зміни навколишнього середовища.</li><li>• Поточні зусилля для поліпшення додатку.</li></ul>
Експериментальне макетування	<ul style="list-style-type: none"><li>• Використовується для перевірки гіпотез.</li><li>• Спробувати технічні рішення для задоволення вимог.</li><li>• Обговорення технічних питань та питання зручності користування.</li><li>• Збереження досвід про придатність і можливості дизайну чи реалізації.</li></ul>
Дослідне макетування	<ul style="list-style-type: none"><li>• Використовується, коли проблема не чітко з'ясована.</li><li>• Експрес-розроблення, як щось повинно працювати і виглядати.</li><li>• Дизайн розвідка різними рішення.</li><li>• Зрозуміти потреби користувачів і завдання, і як вони можуть бути підтримані інформаційними технологіями.</li><li>• Виявити ідеї та сприяння співробітництву й взаємодія між зацікавленими сторонами.</li></ul>

Низько деталізовані прототипи повинні використовуватися для вивчення проекту простору інтерфейсу. Вони не вимагають особливих зусиль і дозволяють швидко порівняння альтернатив. Таким чином, вони підтримують пріоритети проектних альтернатив і звужуються дизайн простору. Хоча прості

конструкції можуть втілювати рішення проектних проблем і забезпечують початкову точку для обговорення, але вони занадто схематичні й занадто неточні, щоб дати подальші вказівки для розробників. Крім того, їм не вистачає технічної естетики.[2]

Таблиця 2.2

### Переваги і недоліки низько і високо деталізованих прототипів

Тип	Переваги	Недоліки
Низько деталізовані	<ul style="list-style-type: none"> <li>• Менше часу і низька вартість.</li> <li>• Проаналізувати різні концепції дизайну.</li> <li>• Пристрій зв'язку.</li> <li>• Проблеми розташування екранних форм.</li> <li>• Визначити потреби ринку.</li> <li>• Доказ правильності концепції.</li> </ul>	<ul style="list-style-type: none"> <li>• Обмежена корисність для тестів зручності користування.</li> <li>• Погана детальна специфікація для кодування.</li> <li>• Навігаційні та потокові обмеження.</li> <li>• Обмежена перевірка помилок.</li> <li>• Обмежене застосування після визначення вимог.</li> </ul>
Високо деталізовані	<ul style="list-style-type: none"> <li>• Часткова або повна функціональність.</li> <li>• Повністю інтерактивний.</li> <li>• Використання для дослідження та випробування.</li> <li>• Вигляд і відчуття кінцевого продукту.</li> <li>• Специфікація реального інтерфейсу користувача.</li> <li>• Маркетинг та інструмент продажів.</li> </ul>	<ul style="list-style-type: none"> <li>• Багато часу, щоб створити.</li> <li>• Найдорожче у розробці</li> <li>• Неефективні для доказу правильності концепції конструкцій.</li> <li>• Приховує від користувачів основні недоліки представлення.</li> <li>• Є відчуття, що це реальна версія інтерфейсів.</li> </ul>

Таблиця 2.3

### Переваги і недоліки низько і високо деталізованих прототипів

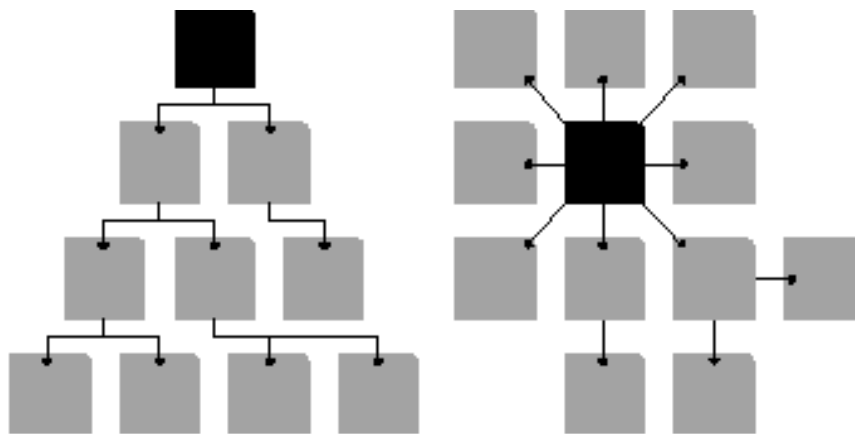
Гнучке моделювання	Проектування з врахуванням зручності використання
<ul style="list-style-type: none"> <li>• Історія користувача (User Story)</li> <li>• Ситуації користувача (User Cases)</li> <li>• Сценарій використання (Usage Scenarios)</li> <li>• Технічні вимоги (Technical Requirements)</li> <li>• Архітектурний стержень (Architectural Spike)</li> </ul>	<ul style="list-style-type: none"> <li>• Моделі ролей (User Roles, User Role Maps)</li> <li>• Моделі задач (User Cases / Maps, Scenarios)</li> <li>• Патерни інтерфейсу (UI Patterns)</li> <li>• Низько та високо деталізовані прототипи (Low/High-Fidelity Prototypes)</li> <li>• Зручність використання та Цілі досвіду користувача (Usability &amp; User Experience Goals)</li> </ul>

### Розробка сценаріїв діалогових систем користувача

Сценарій використання, варіант використання, прецедент або ж користувацький сценарій (англ. Use Case) – у розробці програмного забезпечення та системному проектуванні це опис поведінки системи, яким вона відповідає на зовнішні запити. Іншими словами, сценарій використання описує, «хто» і «що» може зробити з розглянутої системою. Методика сценаріїв використання застосовується для виявлення вимог до поведінки системи, відомих також як функціональні вимоги.

У системному проектуванні сценарії використання застосовуються на більш високому рівні ніж у при розробці програмного забезпечення, часто представляючи мети зацікавлених осіб або місії.

Відправною точкою для виділення основних конкретних об'єктів проектування є розробка загального сценарію діалогової підсистеми. Сценарій діалогу – детальний опис структури та змісту діалогу, що достатній для його реалізації [3].



*Рис. 2.1. Типові структури переходів між інтерфейсними формами програмного забезпечення*

Загальний сценарій є інтегрованим сценарієм локальних підсценаріїв діалогу, що будуються на основі схем рішення задачі в діалозі, і які доповнюються формами представлення вхідних даних та результатів.

Діаграма діалогового сценарію, як правило, використовується для планування модульної структури програми діалогової системи. Так у головний модуль можуть входити програмні фрагменти початкового стану та одного чи декількох наступних станів. На рис.2.2 наведений приклад реалізації діалогових станів та переходів при роботі з програмними моделями графічного редактора принципів електричних схем із вказанням події, що викликали ці переходи. [3]



### Розробка статичного прототипу інтерфейсу користувача

При створенні прототипу найбільш частою помилкою є надмірне наведення глянцю і взагалі прагнення зробити прототип максимально схожим на результуючу систему. Подальші версії прототипу можуть бути естетично більш досконалішими. На основі повної схеми сценарію діалогів, яка описує всю взаємодію користувачів з системою, можна почати розробку окремих прототипів віконних форм програмного забезпечення.

*Найпростіша версія – паперова.* Необхідно намалювати на папері всі екрани і діалогові вікна (роздрукувати відповідні частини схеми). Потрібно тільки переконатися, що всі інтерфейсні елементи виглядають одноманітно і скільки-небудь схоже на реальні. Ця роздруківка і є першим прототипом. На ньому цілком можна тестувати сприйняття системи користувачем і її основну логіку. Користь початкового прототипування на папері полягає у виключній простоті модифікації за результатами тестування на представниках цільової аудиторії.

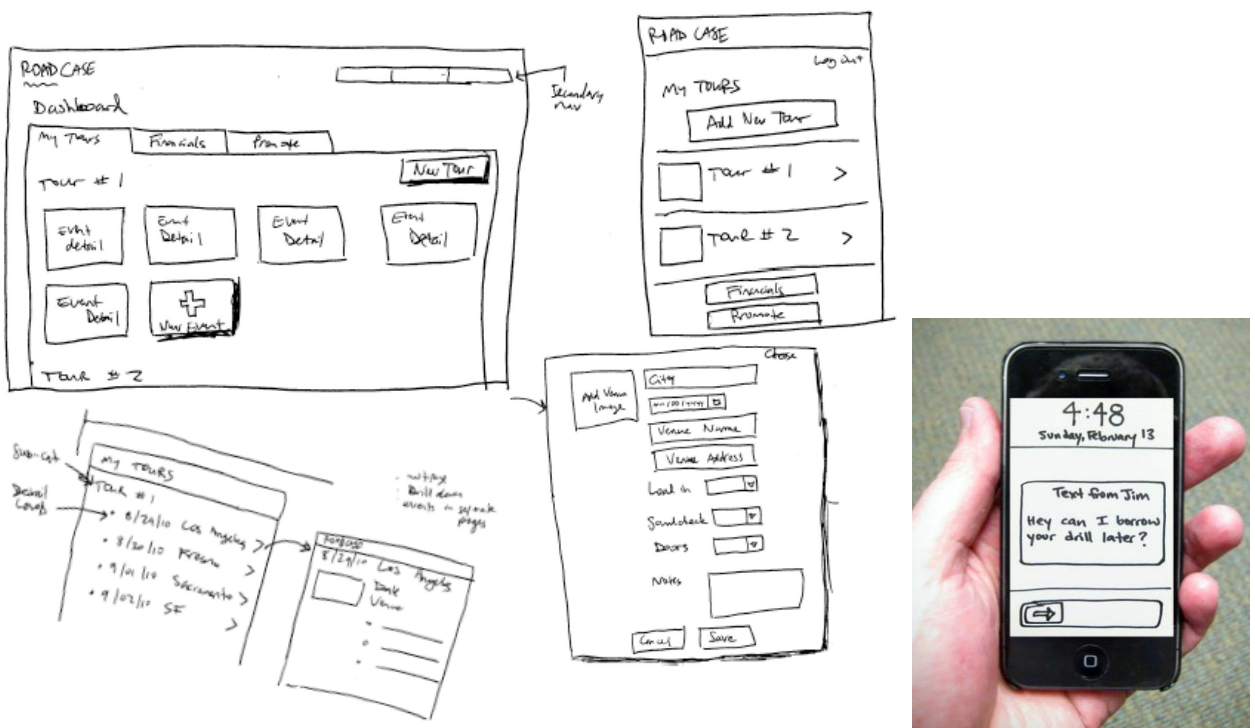


Рис. 2.3 Приклади ескізного «паперового» прототипування [4]

*Друга версія – електронна презентація.* Після вичерпання можливостей «паперової» версії прототипу варто створити нову версію (виправивши, зрозуміло, вже виявлені проблеми). Для цього так само малюється інтерфейс, але вже не на папері, але у будь-якій спеціалізованій програмі у спрощеному креслярському вигляді.



Найпростіший електронний прототип користувацького інтерфейсу є демонстраційним роликом, виконаний в одній з презентаційних програм (наприклад: Microsoft PowerPoint). Кожна екранна форма відповідає окремому слайду, результат натискання кнопок імітується переходами між слайдами. Переходи реалізуються за допомогою організації гіперпосилань. Електронна версія прототипу для користувача інтерфейсу дозволяє тестувати досить складні взаємодії людини з додатком.

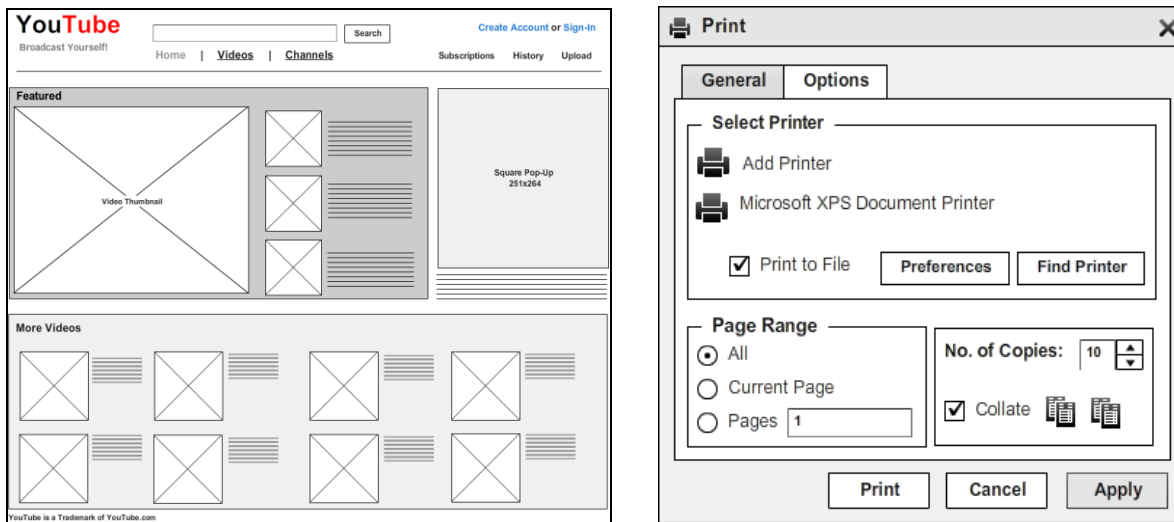


Рис. 2.4 Приклади електронних прототипів інтерфейсу користувача [5]

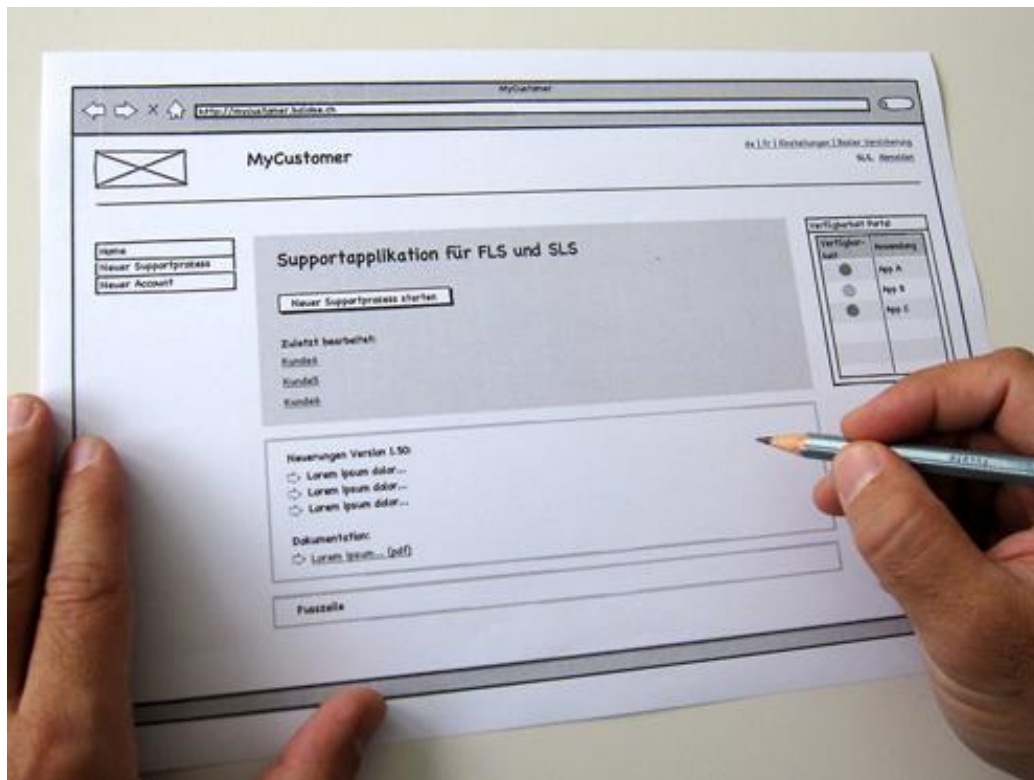


Рис. 2.5 Приклади електронних прототипів інтерфейсу користувача [6]

## Редактори електронних прототипів інтерфейсів користувача

### 1. Microsoft Visio Professional (<http://www.microsoft.com>):

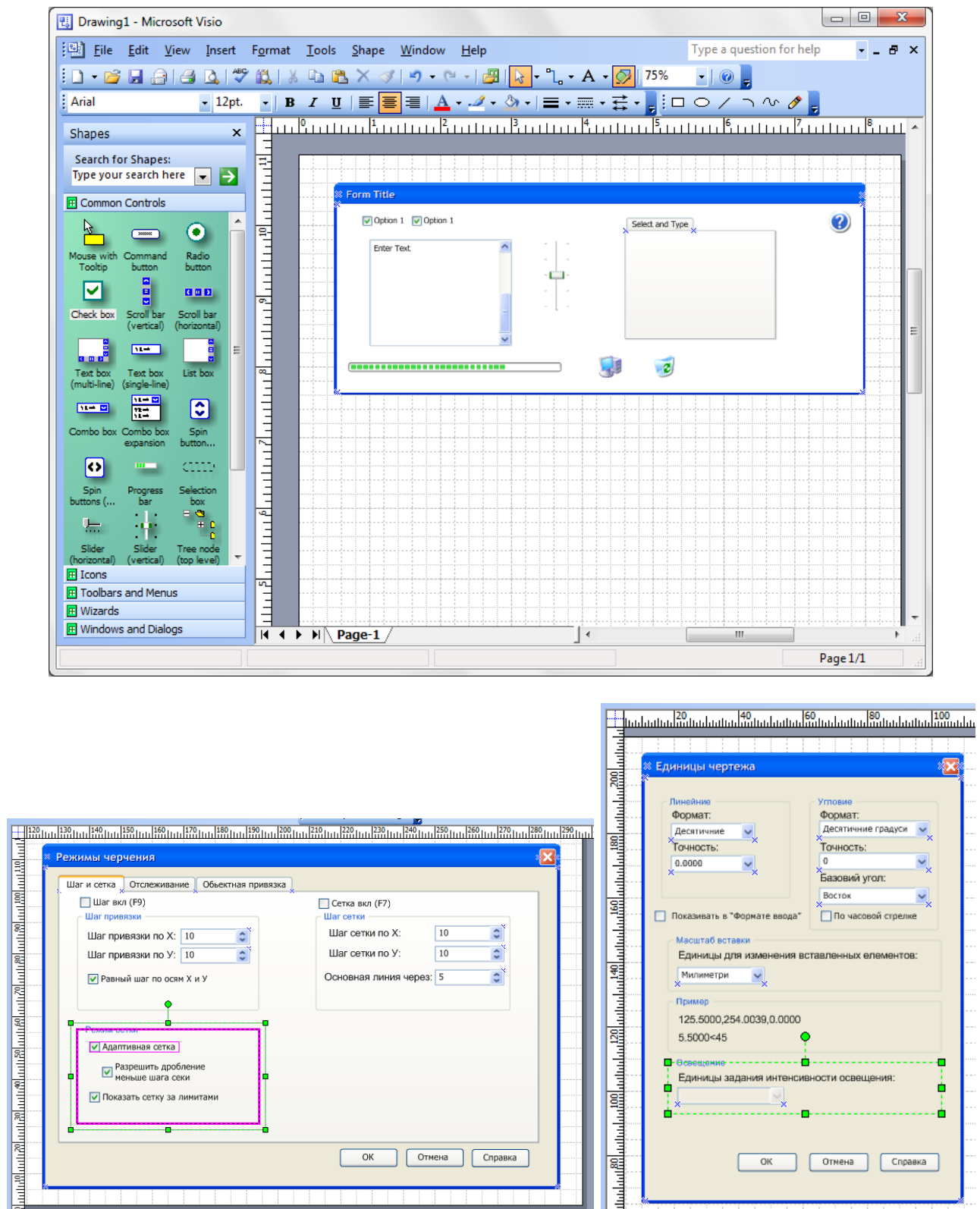


Рис. 2.6 Приклад підготовки прототипів GUI в Microsoft Visio

## 2. Evolus Pencil (<http://pencil.evolus.vn>):

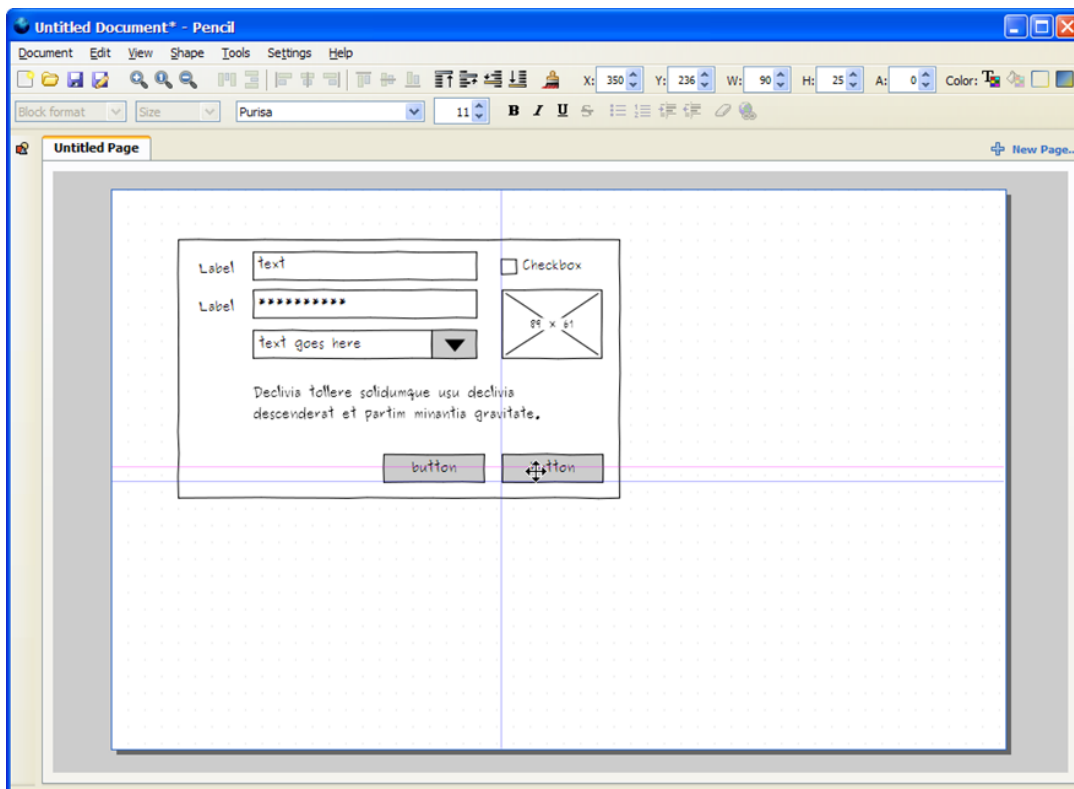


Рис. 2.7 Приклад підготовки прототипів GUI в Evolus Rencil

## 3. Balsamiq Mockups (<http://www.balsamiq.com/products/mockups>):

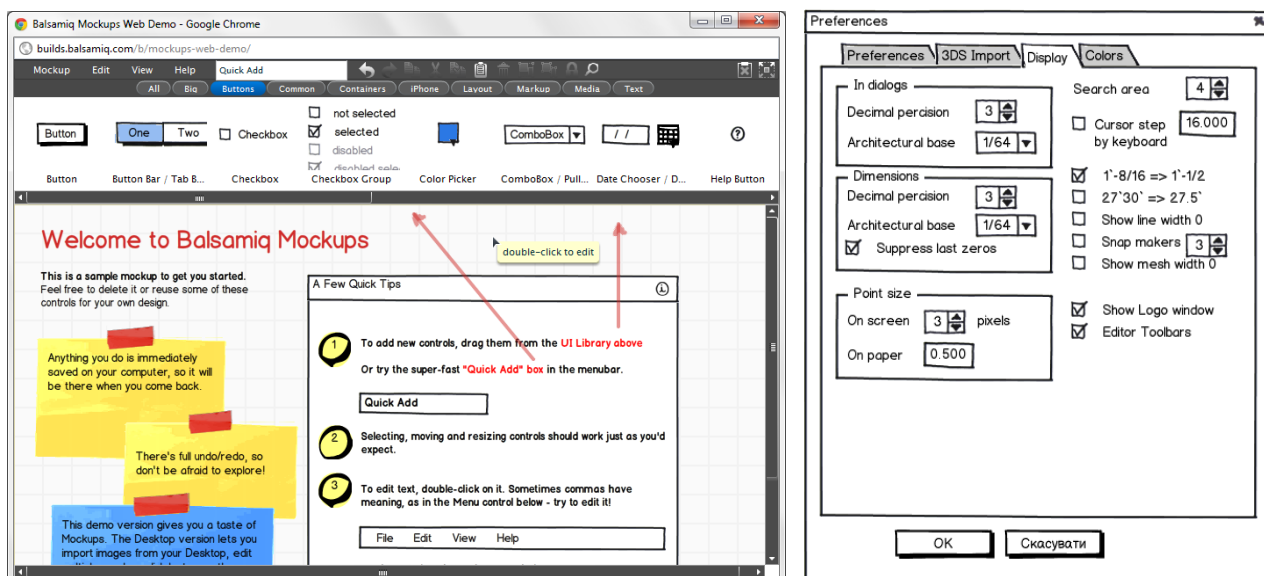


Рис. 2.8 Приклади підготовки прототипів GUI в Balsamiq Mockups

### **Принципи формування електронного динамічного прототипу**

1. Виділяють різні стани екранних форм інтерфейсу. Стан форми визначається поточним станом її елементів управління, таких станів може бути безліч. Наприклад, стани командної кнопки: нейтральне, натиснене, нейтральне до встановленого фокусом введення тощо; стану групи радіокнопок: вибрана перша альтернатива, друга чи інша; стану списків та інших елементів управління. Усі допустимі поєднання станів елементів однієї форми є окремими станами цієї форми. У презентації (Microsoft PowerPoint) – електронному прототипі інтерфейсу – будь-якого з виділених станів екранних форм відповідає окремий слайд.


2. Коли сукупність необхідних слайдів сформована, приступають до складання презентаційного ролика.

3. Ролик не може представляти собою зміну слайдів у фіксованій послідовності, тому що при роботі з програмним додатком користувач може задіяти будь-які доступні на даний момент елементи інтерфейсу.

4. Перехід між слайдами організовують з використанням гіперпосилань. Гіперпосилання тут є зв'язком одного слайда з іншим слайдом, з довільним показом, що містить групу слайдів іншої презентації. Гіперпосилання можуть містити абсолютні або відносні зв'язку.

### **Гіперпосилання для прототипу в презентації Microsoft PowerPoint**

Для створення гіперпосилання на довільний показ або місце в поточній презентації необхідно:

1. Вибрати текст або об'єкт, який повинен представляти гіперпосилання.
2. Натиснути кнопку «Додавання гіперпосилання»  **Hyperlink...** **Ctrl+K**.
3. У новому вікні «Вставка гіперпосилання (Insert Hyperlink)» зліва в області «Зв'язок з (Link to)» вибрати значок місцем у цьому документі.

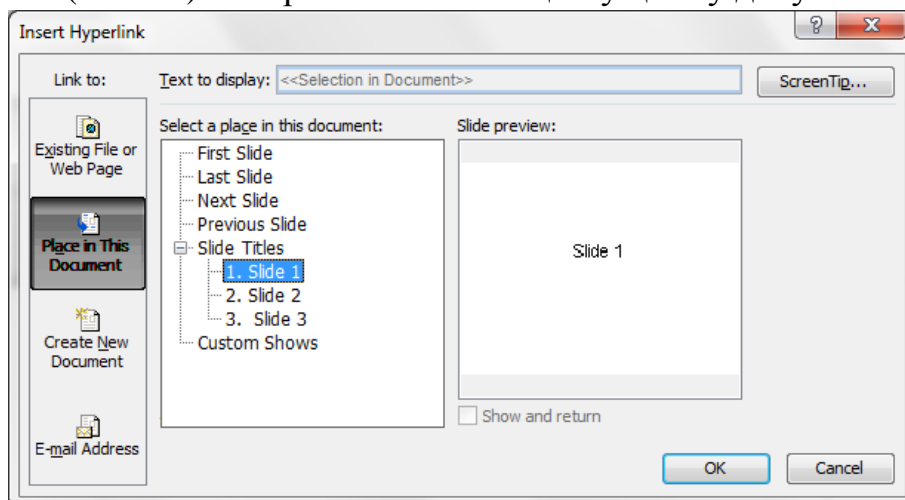



Рис. 2.9 Приклад вікна вставки гіперпосилання

4. Якщо встановлюють зв'язок з іншим слайдом у поточній презентації, вибрати зі списку слайд, до якого потрібно перейти.
5. Якщо встановлюють зв'язок з довільним показом, вибрати зі списку довільний показ, до якого потрібно перейти, потім встановити прапорець показати і повернутися.

*Для створення гіперпосилання на певний слайд в іншій презентації виконують такі дії:*

1. Вибрати текст або об'єкт, який повинен слугувати гіперпосиланням.
2. Натиснути кнопку «Додати гіперпосилання»  Hyperlink... Ctrl+K .
3. В області «Зв'язати з» вибрати пункт наявним файлом.
4. Знайти і вибрати презентацію, яка містить слайд, на який має вказувати гіперпосилання.
5. Натиснути кнопку «Закладка (Bookmark...)» і вибрати заголовок потрібного слайда.

### **3. КОНТРОЛЬНІ ЗАПИТАННЯ**

1. Яка суть ескізного проектування інтерфейсу користувача?
2. Що таке прототип проекту інтерфейсу користувача?
3. На чому зосереджене прототипне проектування інтерфейсу користувача?
4. Що таке сценарій користувацької взаємодії?
5. Які бувають структури переходів сценаріїв інтерфейсних взаємодій?
6. Які бувають версії (форми) прототипу інтерфейсу користувача?
7. Які найбільш відомі спеціалізовані редактори проектування прототипів інтерфейсу користувача?

### **4. ЗАГАЛЬНЕ ЛАБОРАТОРНЕ ЗАВДАННЯ**

1. Ознайомитися з відомостями про підходи, способи, методи та засоби створення прототипів інтерфейсів користувача програмного забезпечення.
2. Отримати у викладача індивідуальне лабораторне завдання на поточний навчальний рік.
3. Налаштувати ПЗ для запису відео з екрану (наприклад: <http://www.camstudio.org/> та кодек CamStudio Lossless Codec). Записати відео з процесом створення прототипів інтерфейсів. Задайте назву файлу з відео

англ. мовою за шаблоном: <рік>\_<група>\_<прізвище>\_lab2\_<назва відео>.avi. (наприклад: 2019\_kn307\_ivanenko\_lab2\_softname\_.avi)

4. Відповідно до індивідуального завдання дослідити складові елементи існуючого інтерфейсу користувача, його вікна, панелі, кнопки та решта складових. Підготувати набір зображень (PrintScreen) вікон інтерфейсу, виділивши при цьому різні стани окремих екранних форм, в яких можуть знаходитися форми в процесі взаємодії користувача з додатком. Файли зберегти у форматі \*.png з назвами за шаблоном: <рік>\_<група>\_<прізвище>\_lab2\_screen\_<xxx>.png (наприклад: 2019\_kn307\_ivanenko\_lab2\_screen\_015.png).
5. Прототипи зробити двома способами, довільним редактором із перелічених:
  - засобами дивитися в Додаток «А» Online UI prototyping Apps;
  - засобами дивитися в Додаток «Б» Desktop UI Prototyping Application
6. На основі переходів між вікнами побудувати схему діалогового сценарію (дивитись приклад у теоретичних відомостях) у редакторі Microsoft Visio. Як на прикладі рис.2.2.
7. Для усіх (PrintScreen) вікон інтерфейсу розробити свої візуальні прототипи (MockUp) у відповідному редакторі прототипування і зберегти файли проекту та графічні файли. Як на прикладі рис.2.4 та рис.2.5.
8. Провести статистичний підрахунок застосованих конструктивних елементів інтерфейсу користувача у відповідних прототипах. Результати оформити у вигляді таблиці. Шаблон показаний у таблиці 4.1:

Таблиця 4.1

**Результати статистичного дослідження інтерфейсів користувача**

Назва елемента інтерфейсу	Загальна кількість, шт $k$	Загальне процентне співвідношення $\frac{k}{N} \cdot 100\%$	Screen №001 кількість, шт	...	Screen № ____ кількість, шт
1	2	3	4	...	...
Button	145	9.385113268608414			
Icon Button	234	15.14563106796117			
Menu	...	...			

...					
...					
Всього (N):	1545	100%			

*Примітка: у таблиці для колонки 1 використати список усіх елементів інтерфейсу з теоретичних відомостей лекцій. Для колонки 3 обчислити процент як співвідношення кількості певного елемента до загальної кількості всіх елементів взятих у процентах. Рекомендується таблицю робити в Microsoft Excel.*



9. Побудувати стовпцеву діаграму на основі таблиці зі всіма підписами параметрів.
10. Сформувати слайди (Microsoft PowerPoint, або іншим подібним редактором) для створення демонстраційного ролика. Кожен слайд відповідає певному стану окремої екранної форми з двох шарів (верхній шар – рисунок прототипу нижній шар зі зміщенням – PrintScreen існуючого інтерфейсу)
11. Оформити звіт про виконання лабораторної роботи у електронному вигляді у форматі Microsoft Word та після перевірки викладачем роздрукувати на папері.

## 5. ВИМОГИ ДО ОФОРМЛЕННЯ ЗВІТУ ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

Звіт про виконання лабораторної роботи оформляється у двох варіантах:

1) В електронному вигляді:

- Звіт у файлі формату Microsoft Word 2003-2019 (\*.doc/\*.docx) з назвою файлу за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab<№>.doc (наприклад:  
2019\_kn307\_ivnenko\_lab2.doc);
- Статистична таблиця у файлі формату Microsoft Excel 2003-2019 (\*.xls / \*.xlsx) з назвою файлу за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab<№>\_table.xls (наприклад:  
2019\_kn307\_ivnenko\_lab2\_table.xls);
- набір файлів копій екрану (PrintScreen). Файли зберегти у форматі \*.png з назвами за шаблоном:



<рік>\_<група>\_<прізвище>\_lab<№>\_screen\_<xxx>.png  
(наприклад: 2019\_kn307\_ivanenko\_lab2\_screen\_015.png);

- Схема сценарію діалогів у файлі формату Microsoft Visio (\*.vsd / \*.vdx) з назвою файлу за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab<№>\_script.vsd (наприклад: 2019\_spk-12\_ivanenko\_lab2\_script.vsd);
- Для обрано редактора із Додатку «А» набір файлів проекту прототипування та їхні графічні еквіваленти у форматі \*.png з назвами за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab<№>\_prototype\_A\_<xxx>.png  
(наприклад: 2019\_kn307\_ivanenko\_lab2\_prototype\_A\_015.png);
- Для обрано редактора із Додатку «А» набір файлів проекту прототипування та їхні графічні еквіваленти у форматі \*.png з назвами за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab<№>\_prototype\_B\_<xxx>.png  
(наприклад: 2019\_kn307\_ivanenko\_lab2\_prototype\_B\_015.png);
- Презентація прототипів сценарію діалогів у файлі формату Microsoft PowerPoint (\*.ppt / \*.pptx) з назвою файлу за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab<№>\_present.ppt (наприклад: 2019\_kn307\_ivanenko\_lab2\_present.vsd);
- Відео файл з кодеком CamStudio Lossless Codec (<http://www.camstudio.org/>) та назвою за шаблоном:  
<рік>\_<група>\_<прізвище>\_lab2\_<назва відео>.avi. (наприклад: 2019\_kn307\_ivanenko\_lab2\_softname.avi)
- усі ці файли необхідно стиснути архіватором \*.zip / \*.rar (шаблон назви файлу архіву – аналогічний).

2) Паперовий вигляд звіту має бути роздрукований на білих аркушах паперу формату А4 і зшитий скріпкою лише з верхньої лівої сторони аркушів.

Після захисту роботи електронні та паперові звіти здаються викладачу для зберігання на кафедрі. Звіт повинен бути підготовлений українською мовою (з елементами англomовної термінології), акуратно та грамотно, з дотриманням правил оформлення ділової документації та державними стандартами.

**Звіт повинен містити такі розділи:**

1. Титульний аркуш з номером лабораторної роботи, назви предмету та темою роботи відповідно до стандартного взірця.
2. Мета виконання лабораторної роботи.



3. Короткі теоретичні відомості із даними, які використані при роботі над індивідуальним завданням.
4. Індивідуальне завдання з детальним формулюванням розв'язуваної задачі.
5. Схема сценарію діалогових взаємодій (приклад: рис.2.2).
6. Набір (двома способами) графічних прототипів усіх розроблених основних, діалогових та допоміжних вікон на основі досліджуваного програмного забезпечення (приклад: рис.2.4 та рис.2.5).
7. Таблиця результатів статистичного розподілу всієї палітри застосованих конструктивних елементів інтерфейсу користувача у створеному прототипі.
8. Стовпцева діаграма на основі таблиці зі всіма підписами параметрів.
9. Короткий аналіз отриманих результатів і висновки до лабораторної роботи.

## 6. ЛІТЕРАТУРА

1. Торрес Р., Практическое руководство по проектированию и разработке пользовательского интерфейса / Торрес Р. // Вильямс, 2002, 400 с., ISBN:5-8459-0367-X
2. Memmel Thomas, Prototyping Corporate User Interfaces - Towards A Visual Specification Of Interactive Systems / Memmel, Thomas; Gundelsweiler, Fredrik; Reiterer, Harald // Proceedings of the 2nd IASTED International Conference on Human Computer Interaction (IASTED-HCI 07), Chamonix, France: Proceedings, Acta Press, Canada, p. 177-182, Mar 2007 [Електронний ресурс] Режим доступу: <http://hci.uni-konstanz.de/downloads/569-070.01Ai.pdf>
3. Маркелов О.Е. Розробка сценаріїв діалогів графічного редактора електричних схем системи схемотехнічного проектування "Micro-PC"/ Маркелов О.Е. // Вісник національного університету "Львівська політехніка" "Радіoeлектроніка та телекомунікації", №443, 2002, с.193-199.
4. Road Case, Sketchy sketch September 22, 2011 [Електронний ресурс] Режим доступу: <http://roadcase.wordpress.com/2011/09/22/sketchy-sketch/>
5. MockFlow Wireframing experience samples, A Produlle Systems (P) Limited, 2012 [Електронний ресурс] Режим доступу: <http://www.mockflow.com/samples/>
6. Usability Engineering, Canoo Support [Електронний ресурс] Режим доступу: <http://www.canoo.com/services/usability/>

7. Гульяев А.К., Проектирование и дизайн пользовательского интерфейса / А.К. Гульяев, В.А. Машин // Корона Принт, 2007, 352 с., ISBN:978-5-7931-0477-7
8. Дженифер Тидвелл, Разработка пользовательских интерфейсов. 2-е издание / Дженифер Тидвелл // Питер, 2011, 480 с., ISBN:978-5-459-00434-2
9. Ажеронок В. А., Разработка управляемого интерфейса / В. А. Ажеронок, А. В. Островерх, М. Г. Радченко, Е. Ю. Хрусталева // 1С-Паблишинг, 2010, 728 с., ISBN: 978-5-9677-1148-0
10. Билл Скотт, Проектирование веб-интерфейсов / Билл Скотт, Тереза Нейл // Символ-Плюс, 2010, 352 с., ISBN 978-5-93286-172-1, 978-0-596-51625-3

## ДОДАТОК «А» ONLINE UI PROTOTYPING APPS

1. Balsamiq Mockups (<http://www.balsamiq.com/products/mockups>)
2. Pidoco (<https://pidoco.com>)
3. Produte Systems Mockflow (<http://www.mockflow.com> )
4. Rudrasoft MockupTiger (<http://www.mockuptiger.com> )
5. Cyberopsis Webdata Proto.IO (<https://proto.io> )
6. Site9 Inc. ProtoShare (<http://www.protoshare.com> )
7. Evolus Pencil (<http://pencil.evolus.vn> )
8. Tool Software UI Mockup (<http://www.toolsoft.net/ui-mockup.html> )
9. ERGOSIGN Technologies Antetype (<http://www.antetype.com> )
10. Endloop Systems iMockups (<http://www.endloop.ca/imockups> )
11. CATCH Screen Architect (<http://www.catchlimited.com> )
12. FluidUI (<http://www.fluidui.com> )
13. Codiqa (<http://codiqa.com> )
14. UserVoice Inc. Moqups (<https://moqups.com> )
15. Some Character LLC Mockingbird (<https://gomockingbird.com> )
16. JGraph Ltd. draw.io (<https://www.draw.io> )
17. Tiggzi (<http://tiggzi.com> )
18. HotGloo (<http://www.hotgloo.com> )
19. MockupBuilder (<http://mockupbuilder.com/App> )
20. iPlotz (<http://iplotz.com> )
21. Maqetta (<http://maqetta.org> )
22. Simulify (<http://www.simulify.com> )
23. JustProto (<http://www.justproto.com> )
24. Lumzy (<http://lumzy.com/app> )
25. Solidify (<http://www.solidifyapp.com> )
26. AndroidMockup (<http://yeblon.com/androidmockup> )
27. iPhoneMockup (<http://iphonemockup.lkmc.ch> )
28. Easel (<https://www.easel.io/demo> )
29. Creately Diagraming (<https://creately.com/app> )

## ДОДАТОК «Б» DESKTOP UI PROTOTYPING APPLICATION

1. Microsoft Visio Professional (<http://office.microsoft.com/en-us/visio> )
2. Axure Software Solutions (<http://www.axure.com> )
3. MockupScreens (<http://www.mockupscreens.com> )
4. Evolus Pencil (<http://pencil.evolus.vn> )
5. AppSketcher <http://www.appsketcher.com> )
6. FlairBuilder (<http://www.flairbuilder.com> )
7. PowerMockup <http://www.powermockup.com> /
8. Indigo Studio <http://www.infragistics.com/products/indigo-studio>
9. PowerStory <http://www.power-story.com/>
10. SoftandGUI <http://www.softandgui.co.uk/>
11. Prototyper Pro Edition <http://www.justinmind.com/prototyper/download>
12. ForeUI <http://www.foreui.com/>
13. inPreso <http://www.inpreso.com/inpresoscreens/>
14. Wireframesketcher <http://wireframesketcher.com/>
15. Irise Studio [http://www.irise.com/products/irise\\_studio](http://www.irise.com/products/irise_studio)
16. GUI Machine <http://guimachine.ru/>
17. DesinerVista <http://www.designervista.com/download.php>
18. MockupDesigner <http://mockupdesigner.codeplex.com/>
19. GUI Design Studio <http://www.carettasoftware.com>
20. Serena Prototype Composer  
<http://www.serena.com/index.php/en/products/prototype-composer/>
21. FlairBuilder <http://www.flairbuilder.com>
22. Pencil - addon for Firefox browser <https://addons.mozilla.org/en-US/firefox/addon/pencil/>
23. Antetype for MacOS X <http://www.antetype.com/>
24. DENIM <http://dub.washington.edu:2007/denim/>
25. Xojo <http://xojo.com/>

НАВЧАЛЬНЕ ВИДАННЯ

ПРОТОТИПУВАННЯ КОРИСТУВАЦЬКИХ ІНТЕРФЕЙСІВ ПРОГРАМНОГО  
ЗАБЕЗПЕЧЕННЯ

**МЕТОДИЧНІ ВКАЗІВКИ**

до лабораторної роботи № 2  
з курсу «Людино-машинна взаємодія»  
для студентів бакалаврів напрямку 6.122 – Комп'ютерні науки

Укладач: Маркелов Олександр Едуардович

*Редактор*

*Комп'ютерне верстання*

Друк на різнографі.  
Наклад 15 прим.

Поліграфічний центр  
Видання кафедри «Системи автоматизованого проектування»,  
Інститут комп'ютерних наук та інформаційних технологій,  
Національного університету “Львівська політехніка”  
*вул. С. Бандери, 12, навч. корп. 4, кім. 320, кім. 324, 79013, Львів*