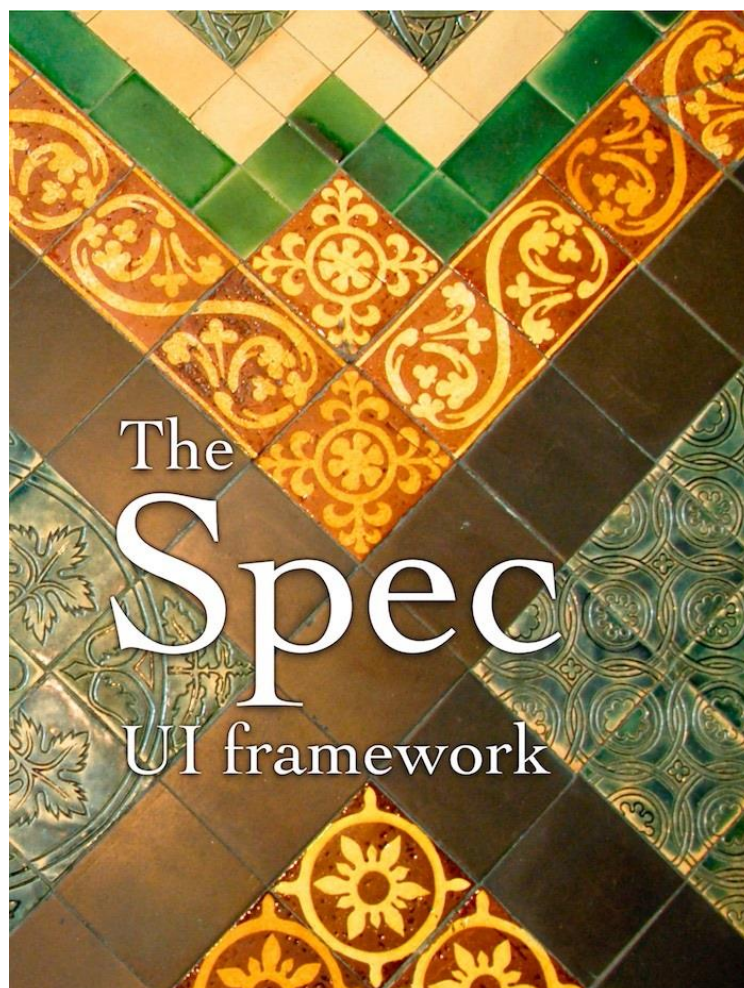




Побудова застосунків засобами Spec 2.0

Компонентно орієнтована розробка GUI



Application Building with Spec 2.0

K. De Hondt, S. Ducasse with S. Jordan-Montaña and
E. Lorenzano

October 26, 2024

BoD – Books on Demand

Побудова застосунків засобами Spec 2.0

К. Де Хондт, С. Дюкас разом з С. Джордан-Монтаньо та
Е. Лоренцано

Переклад українською з доповненнями
Сергій Ярошко

ЛНУ ім. Івана Франка
2025

УДК 004.432.2

X 77

Перекладено за оригіналом:

K. De Hondt, S. Ducasse with S. Jordan Montaña and E. Lorenzano
Application Building with Spec 2.0

Переклад:

Ярошко С. А. – канд. фіз.-мат. наук, доцент
(Львівський національний університет імені Івана Франка)

Перекладено з дотриманням умов ліцензії Creative Commons Attribution-ShareAlike 3.0 Unported.



Рецензенти:

Іванчук Я. В. – д-р техн. наук, професор
(Вінницький національний технічний університет);

Мица О. В. – д-р техн. наук, професор
(Ужгородський національний університет);

Яремко Г. В. – канд. пед. наук, доцент
(Національний університет «Львівська політехніка»)

Рекомендовано до друку

Вченою радою Львівського національного університету імені Івана Франка.

Протокол № 82/5 від 01.05.2025 року

Коен Де Хондт

X 77 Побудова застосунків засобами Spec 2.0/ К. Де Хондт, С. Дюкас [та ін.] ; пер. з англ. з допов. С. Ярошка. – Електрон. вид. – Львів : ЛНУ ім. Івана Франка, 2025. – 240 с.

ISBN 978-617-10-1094-9 (електрон. вид.)

Програмний каркас Spec призначено для побудови віконного інтерфейсу користувача застосунків, розроблених у середовищі програмування Pharo, сучасному втіленні класичної системи Smalltalk. Програмування у Spec можна назвати компонентно-орієнтованим, оскільки кожен візуальний компонент може стати і самостійним інтерфейсом, і частиною більшого компонента. Послідовно викладено правила побудови застосунку засобами Spec, описано демонстратори (стандартні і збудовані користувачем), макети, стилі, команди, перенесення даних, вікна та діалоги. Наведено багато прикладів конкретних застосунків.

Для здобувачів вищої освіти галузі знань F Інформаційні технології і всіх, хто цікавиться сучасними технологіями програмування.

УДК 004.432.2

ISBN 978-617-10-1094-9 (електрон. вид.)

© Хондт К., Дюкас С., [та ін.], 2024

© Ярошко Сергій, переклад, 2025

© Львівський національний університет імені Івана Франка, 2025

1. Вступ.....	1
1.1. Повторне використання інтерфейсів.....	1
1.2. Spes 2.0.....	3
1.3. Програмний код прикладів.....	4
1.4. Подяки	5
1.5. Від перекладача	5
I. Увесь Spes в одному прикладі.....	6
2. Маленький приклад на 10 хвилин.....	7
2.1. Вікно для опитування клієнтів.....	7
2.2. Створіть клас для інтерфейсу користувача	7
2.3. Створення та налаштування вкладених демонстраторів	7
2.4. Визначення заголовка та розміру вікна, відкривання та закривання.....	10
2.5. Підсумки розділу	11
3. Більша частина Spes в одному прикладі.....	12
3.1. Застосунок	12
3.2. Базова модель фільму.....	12
3.3. Список фільмів.....	13
3.4. Заповнення списку фільмів	14
3.5. Відкривання демонстраторів за допомогою застосунку	15
3.6. Покращення вигляду вікна.....	15
3.7. Застосунок керує піктограмами.....	16
3.8. Демонстратор фільму	16
3.9. Покращення вигляду демонстратора фільму	18
3.10. Відкривання демонстратора фільму в вікні модального діалогу	19
3.11. Налаштування модального діалогу.....	20
3.12. Виклик діалогу	20
3.13. Вбудовування демонстратора фільму у демонстратор списку фільмів.....	21
3.14. Визначення взаємодії компонентів.....	22
3.15. Тестування графічного інтерфейсу користувача застосунку	23
3.16. Більше тестів	25
3.17. Заміна макета.....	26
3.18. Використання перенесень.....	27

3.19.	Використання стилів для оформлення застосунку	28
3.20.	Підсумки розділу	30
II.	Основи Spec.....	31
4.	Кількома словами про ядро Spec	32
4.1.	Огляд архітектури Spec	32
4.2.	Огляд архітектури ядра Spec	33
4.3.	Демонстратори	33
4.4.	Застосунок.....	34
4.5.	Налаштування застосунку	34
4.6.	Макети.....	35
4.7.	Стили та таблиці стилів	37
4.8.	Поведінка демонстратора.....	37
4.9.	Підсумки розділу	38
5.	Тестування застосунків Spec.....	39
5.1.	Тестування демонстраторів	39
5.2.	Приклад для користувача Spec	40
5.3.	Тести	44
5.4.	Тестування застосунку	48
5.5.	Відомі обмеження та підсумки розділу	49
6.	Подвійне призначення демонстраторів: модель даних і взаємодія	50
6.1.	Про демонстратор з моделлю даних.....	50
6.2.	Приклад з <i>SpPresenter</i>	50
6.3.	<i>SpPresenter</i> VS <i>SpPresenterWithModel</i>	51
6.4.	Приклад з <i>SpPresenterWithModel</i>	52
6.5.	Побудова інтерфейсу користувача: модель подання	53
6.6.	Метод <i>initializePresenters</i>	54
6.7.	Метод <i>connectPresenters</i>	55
6.8.	Метод <i>defaultLayout</i>	56
6.9.	Підсумки розділу	56
7.	Повторне використання і композиція в ділі.....	57
7.1.	Початкові вимоги	57
7.2.	Створення інтерфейсу користувача для використання як компонента.....	58
7.3.	Підтримка повторного використання.....	59
7.4.	Компонування двох базових демонстраторів у новий компонент	59
7.5.	Інспектування «живих» компонентів.....	61
7.6.	Написання тестів	62

7.7.	Управління трьома компонентами та їхньою взаємодією	62
7.8.	Використання різних макетів.....	64
7.9.	Розширення API	65
7.10.	Зміна макета використаного компонента	66
7.11.	Заміна макета.....	67
7.12.	Міркування щодо загальнодоступного API конфігурування.....	68
7.13.	Нові шаблони проти старих.....	68
7.14.	Підсумки розділу	69
8.	Списки, таблиці та дерева	70
8.1.	Списки.....	70
8.2.	Налаштування способу відображення елементів	70
8.3.	Оздоблення елементів списку	71
8.4.	Про одиничний або множинний вибір	72
8.5.	Перетягування і скидання.....	72
8.6.	Активування клацанням	73
8.7.	Списки з фільтром	73
8.8.	Фільтровані списки з можливістю позначення	75
8.9.	Списки компонентів	76
8.10.	Дерева	76
8.11.	Таблиці	78
8.12.	Перша таблиця	79
8.13.	Заголовки, здатні сортувати	79
8.14.	Редаговані таблиці	80
8.15.	Ієрархічні таблиці	81
8.16.	Підсумки розділу	82
9.	Керування вікнами	83
9.1.	Робочий приклад.....	83
9.2.	Відкривання вікна або панелі діалогу.....	84
9.3.	Запобігання закриттю вікна	85
9.4.	Дії під час закривання вікна	86
9.5.	Розмір і оформлення вікна.....	86
9.6.	Отримання значень з діалогу	89
9.7.	Стандартні демонстратори модальних діалогів	90
9.8.	Розміщення демонстратора у вікні діалогу	91
9.9.	Налаштування фокусу введення клавіатури.....	91
9.10.	Дії під час відкривання вікна	92
9.11.	Оновлене API налаштування вікна	94
9.12.	Запам'ятовування зміненого розміру вікна.....	94

9.13. Підсумки розділу	95
10. Макети	96
10.1. Нагадування про основний принцип.....	96
10.2. Робочий приклад.....	96
10.3. Послідовний макет (<i>SpBoxLayout</i> і <i>SpBoxConstraints</i>).....	97
10.4. Вирівнювання елементів послідовного макета	99
10.5. Приклад вирівнювання	99
10.6. Вирівнювання в горизонтальному макеті	101
10.7. Складніші макети	102
10.8. Приготування прикладу для повторного використання.....	104
10.9. Відкривання з певним макетом.....	105
10.10. Ліпша архітектура коду.....	105
10.11. Вибір макета для демонстратора, який використовують повторно.....	106
10.12. Альтернативний спосіб вибору макета	107
10.13. Динамічна зміна макета.....	108
10.14. Прямокутна сітка (<i>SpGridLayout</i>).....	108
10.15. Розділений макет (<i>SpPanedLayout</i>).....	110
10.16. Макет з накладанням (<i>SpOverlayLayout</i>).....	111
10.17. Підсумки розділу	113
11. Динамічні демонстратори.....	114
11.1. Макети такі ж прості як об'єкти	114
11.2. Динамічне додавання кнопок.....	116
11.3. Визначення методів додавання/вилучення кнопок	117
11.4. Побудова невеликого динамічного оглядача.....	117
11.5. Візуальне розташування компонентів	119
11.6. Налаштування взаємодії	120
11.7. Перемикання режимів редагування/тільки для читання	121
11.8. Про перебудову макета	122
11.9. Підсумки розділу	123
12. Конкретний випадок: поштовий застосунок	124
12.1. Моделі.....	124
12.2. Електронний лист.....	125
12.3. Папка для листів	126
12.4. Обліковий запис.....	127
12.5. Демонстратори	129
12.6. Демонстратор листа	129
12.7. Демонстратор інформаційного повідомлення	131

12.8.	Демонстратор-обгортка	131
12.9.	Демонстратор облікового запису	132
12.10.	Демонстратор поштового клієнта.....	134
12.11.	Увесь застосунок	136
12.12.	Підсумки розділу	137
13.	Рядок меню, панель інструментів, рядок статусу і контекстні меню	138
13.1.	Додавання до вікна рядка меню.....	138
13.2.	Реалізація команд підменю «Message»	139
13.3.	Встановлення гарячих клавіш	141
13.4.	Визначення дій.....	141
13.5.	Додавання панелі інструментів.....	143
13.6.	Керування доступністю кнопок	145
13.7.	Додавання рядка стану	146
13.8.	Додавання до демонстратора контекстного меню	151
13.9.	Блоки перевірки доступності	152
13.10.	Підсумки розділу	155
14.	Використання портів і перенесень.....	156
14.1.	Що таке перенесення?	156
14.2.	Простий приклад.....	156
14.3.	Основа перенесення.....	157
14.4.	Перетворення перенесеного об'єкта	158
14.5.	Виконання перенесень без вхідного порту	159
14.6.	Дії після перенесення	159
14.7.	Наявні класи портів	161
14.8.	Порти та вкладені демонстратори.....	161
14.9.	Складніший приклад.....	161
14.10.	Ще одна видозміна	163
14.11.	Підсумки розділу	163
15.	Стилізація застосунку	164
15.1.	Кількома словами	164
15.2.	Як працюють стилі?	165
15.3.	Таблиці стилів.....	165
15.4.	Оголошення стилю.....	165
15.5.	Приклади таблиць стилів	166
15.6.	Анатомія стилю	167
15.7.	Змінні середовища.....	167
15.8.	Зміни на рівні кореня.....	168
15.9.	Визначення стилів застосунку	168

15.10. Застосування стилів.....	169
15.11. Динамічне застосування стилів.....	171
15.12. Підсумки розділу	172
16. Використання Athens і Roassal у Spec	173
16.1. Вступ	173
16.2. Безпосередня інтеграція Athens зі Spec	174
16.3. Інтеграція Roassal зі Spec.....	175
16.4. Клас <i>SpRoassalPresenter</i>	176
16.5. Вітання у Athens через об'єкти Morphic.....	177
16.6. Опрацювання зміни розміру	178
16.7. Використання морф зі Spec.....	178
16.8. Підсумки розділу	179
17. Налаштування Інспектора	180
17.1. Створення власних вкладок	180
17.2. Додавання вкладки з текстом	181
17.3. Вкладка з таблицею	182
17.4. Умова активації вкладки	182
17.5. Додавання необробленого подання елемента колекції	183
17.6. Вилучення панелі інтерпретатора	183
17.7. Додавання діаграм Roassal	184
17.8. Підсумки розділу	185
III. Використання команд. Приклади використання Spec.....	186
18. Commander: потужний і простий програмний каркас для команд.....	187
18.1. Команди.....	187
18.2. Визначення команд.....	188
18.3. Оголошення спільного надкласу команд	188
18.4. Додавання головних команд.....	188
18.5. Додавання команд-заповнювачів.....	190
18.6. Перетворення команд на пункти меню.....	191
18.7. Використання <i>fillWith:</i>	192
18.8. Керування піктограмами та клавіатурними скороченнями	193
18.9. Створення головного меню	193
18.10. Впровадження груп.....	195
18.11. Розширення меню.....	196
18.12. Оголошення розширення.....	198
18.13. Створення панелі інструментів	199
18.14. Підсумки розділу	202

19. Модель світлофора	203
19.1. Вимоги до програми	203
19.2. Графічні елементи	204
19.3. Модель світлофора	205
19.4. Панель керування світлофором	209
19.5. Вигляд світлофора та застосунок	210
19.6. Розширення можливостей застосунку	213
19.7. Підсумки до розділу	216
20. Група залежних перемикачів – демонстратор для повторного використання	217
20.1. Формулювання завдання	217
20.2. Оголошення класу <i>SpRadioGroupPresenter</i>	218
20.3. Створення вкладених демонстраторів	218
20.4. Налаштування поведінки	219
20.5. Макет групи	219
20.6. Доповнення інтерфейсу <i>SpRadioGroupPresenter</i>	220
20.7. Динамічна зміна макета та інші удосконалення	221
20.8. Приклад використання <i>SpRadioGroupPresenter</i>	224
20.9. Підсумки до розділу	226

Application	<i>Застосунок</i>	Комп'ютерна програма з віконним графічним інтерфейсом користувача
Debugger	<i>Налагоджувач</i>	Програма, інструмент для дослідження коду, що виконується. У Phago можна використовувати для написання програм на льоту
File Browser	<i>Оглядач файлів</i>	Інструмент для доступу до файлової системи комп'ютера, має можливості завантаження, швидкого перегляду файлів
Finder	<i>Шукач</i>	Програма, інструмент для відшукування методів, класів, програмного коду, анотацій, прикладів. Уміє шукати за зразком перетворень
Iceberg	<i>Менеджер пакетів</i>	Система контролю версій для взаємодії з Git
Image	<i>Образ системи</i>	Колекція всіх об'єктів системи, завантажується у віртуальну машину
Inspector	<i>Інспектор об'єктів</i>	Програма, інструмент, який дає змогу заглянути всередину об'єкта, вивчити та змінити його стан
Layout	<i>Макет</i>	Клас, який керує розташування візуальних компонентів у вікні застосунку
Morph	<i>Морфа</i>	Видимий елемент інтерфейсу користувача
Playground	<i>Пісочниця</i>	Вікно для безпечного виконання фрагментів коду, інтегроване з Інспектором об'єктів
Presenter	<i>Демонстратор</i>	Клас, що відображає візуальний компонент, частину графічного інтерфейсу користувача, реалізує логіку його поведінки
Spotter	<i>Навідник</i>	Вікно діалогу для швидкого відшукування будь-чого в системі
System Browser	<i>Оглядач класів, Системний оглядач</i>	Головний інструмент системи Phago, який надає доступ до всіх пакетів класів, їхніх методів, дає змогу оголошувати нові класи та редагувати наявні
Transcript	<i>Консоль системи</i>	Текстове вікно, яке можна використовувати для виведення тестових повідомлень від об'єктів, публікує повідомлення про помилки під час інсталяції пакетів
World Menu	<i>Головне меню</i>	Phago – окремий світ на комп'ютері, тому його головне меню називається «Світ»

Електронне навчальне видання

К. Де Хондт, С. Дюкас разом з
С. Джордан-Монтаньо та Е. Лоренцано

Побудова застосунків засобами Spres 2.0

Переклад українською з доповненнями
Сергія ЯРОШКА

Редактор *Н. Плиса*
Комп'ютерне верстання *С. Ярошко*

Формат 60×84/8. Ум. друк. арк. 27,7. Зам. 21Е.

Видавець та виготовлювач:
Львівський національний університет імені Івана Франка,
вул. Університетська, 1, Львів, 79000
Свідоцтво про внесення суб'єкта
видавничої справи до Державного реєстру видавців, виготівників
і розповсюджувачів видавничої продукції.
Серія ДК № 3059 від 13.12.2007 р.