

(М.П.)

«Затверджую»

Проректор з НМР д.т.н., проф. *Тонкачев Г.М.*«04» 06 2020 р. *[Підпис]*Зав. відділу докторантури і аспірантури
к.т.н., доц. *Михайловський Д.В.*«03» 06 2020 р. *[Підпис]*Зав. кафедрою інформаційних технологій
д.т.н., проф. *Цюцора С.В.*«02» 06 2020 р. *[Підпис]*

КАРТА ДИСЦИПЛІНИ (СИЛАБУС)

1) НАЗВА ДИСЦИПЛІНИ: СУЧАСНІ МЕТОДОЛОГІЇ ПРОЕКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	2) Шифр за ОНП: ДВ.03
3) Карта дисципліни дієсна протягом навчального року: 2020/2021	
4) Освітній рівень: третій рівень вищої освіти (доктор філософії)	
5) Форма навчання: денна, заочна	
6) Галузь знань: 12 «Інформаційні технології»	
7) Спеціальність: 122 «Комп'ютерні науки»	
8) Компонента спеціальності: вибіркові	
9) Семестр: III	
10) Цикл дисципліни: дисципліни вільного вибору здобувача	
11) Викладач (розробник карти): доцент, к.т.н. Цюцора М.І., доцент, к.т.н. Київська К.І.	
12) Мова навчання: українська	
13) Необхідні ввідні дисципліни: (що треба вивчити, щоб слухати цей курс) «Міжнародні стандарти з управління інформаційними системами», «Методи та моделі формалізованого представлення результатів наукових досліджень та їх аналізу», «Сучасний стан наукових досліджень в галузі управління ІТ проектами»	
14) Мета курсу: с засвоєння сучасних методологій проектування, розуміння процесу побудови програмного забезпечення, оволодіння методологічними та методичними основами побудови архітектури програмних застосунків в галузі інформаційних технологій.	

15) Результати навчання:

№	Програмний результат навчання	Метод перевірки навчального ефекту	Форма проведення занять	Посилання на програмні компетентності
1.	ПРН03. Здатність орієнтуватися в постановці завдання і визначати, яким чином слід шукати засоби її рішення.	Обговорення під час занять, тематичне дослідження, контрольна робота	Практичні заняття	КК ЗК06 ФК01, ФК04 ФК05

2.	ПРН10. Вміння готувати презентації, оформляти результати досліджень у вигляді статей і доповідей на науково-технічних конференціях.	Обговорення під час занять, тематичне дослідження, контрольна робота	Практичні заняття	ІК ЗК06 ФК01 ФК04 ФК05
3.	ПРН11. Володіння сучасними інформаційними технологіями.	Обговорення під час занять, тематичне дослідження, контрольна робота	Практичні заняття	ІК ЗК06 ФК01 ФК04 ФК05

16) Форми занять та їх тривалість (кількість годин)

Лекція	Практичне заняття	Лабораторні заняття	Курсовий проект/ курсова робота РГР/Контрольна робота	Самостійні робота здобувача
	50		Контрольна робота	100

Зміст: (окремо для кожної форми занять – Л/Пр/Лаб/ КР/СРС)

Лекція – лекційні заняття відсутні

Практичне:

Тема 1. ПОВТОРЕННЯ ЗАГАЛЬНИХ ПРИНЦИПІВ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Проектування класів. Метод функціональної декомпозиції Поняття класу, властивості, методи, модифікатори доступу до даних класів. Відносини між класами: клієнти та нащадки. Конструктори та деструктори. Побудова діаграми класів. Розширення функціоналу батьківського класу: перевантаження, приховування, перевизначення. Спадкування, інкапсуляція та поліморфізм. Множинне наслідування. Інтерфейси. Класи з подіями. Коректна робота з подіями.

Тема 2. ПРИНЦИПИ ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ АНАЛІЗУ ПРЕДМЕТНОЇ ОБЛАСТІ

Поняття проектування програмного забезпечення на основі аналізу предметної області. Метод аналізу спільності/змінності. Визначення аналізу предметної області, еквівалентність понять сутність предметної області та класу об'єктів. Загальні принципи проектування. Типові причин перепроєктування. Якісні характеристики коду: гнучкість, переносність, розширюваність. Механізми повторного використання коду: спадкування, композиція, делегування обов'язків. Плюси і мінуси описаних механізмів, випадки найефективнішого використання.

Тема 3. ВВЕДЕННЯ В ШАБЛони ПРОЕКТУВАННЯ

Передумови виникнення шаблонів проектування. Забезпечення багаторазового використання коду. Класифікація патернів. Визначення, історія та причини виникнення патернів проектування: патерни проектування класів/об'єктів (GoF), архітектурні системні патерни, патерни інтеграції корпоративних інформаційних систем. Класифікація патернів проектування класів/об'єктів: породжувальні, структурні та поведінкові патерни. Складові частини патернів проектування. Послідовність дій по розробці програмного забезпечення із використанням патернів проектування.

Тема 4. ПОРОДЖУВАЛЬНІ ПАТЕРНИ

Породжувальні патерни – загальне визначення, призначення, способи використання, випадки застосування: Абстрактна Фабрика (Abstract Factory), Прототип (Prototype), Будівельник (Builder), Одинак (Singleton), Фабричний Метод (Factory Method): визначення, мета використання, переваги та недоліки використання, застосовність, діаграма класів, учасники шаблонів, програмна реалізація. Загальні висновки стосовно областей та способів використання породжувальних патернів, порівняльна характеристика. Приклади практичного застосування породжувальних патернів.

Тема 5. ЗАГАЛЬНА ХАРАКТЕРИСТИКА СТРУКТУРНИХ ПАТЕРНІВ

Загальний опис структурних патернів. Структурні патерни рівня класів та об'єктів. Програмні механізми, що використовуються при застосуванні структурних патернів. Спільність, різниця, способи реалізації. Випадки та приклади застосування. Патерн Адаптер (Adapter), його різновиди, призначення, проблемні області застосування, переваги та недоліки використання, діаграма класів, учасники шаблону, відношення учасників, результати використання, програмна реалізація, відомі застосування.

Тема 6. СТРУКТУРНІ ПАТЕРНИ

Патерни Міст (Bridge), Компонувальник (Composite), Фасад (Facade), Декоратор (Decorator), Заступник (Proxy), Легковаговик (Flyweight):

Призначення, синонімічні назви, мотивація використання, застосовність, переваги та недоліки використання, діаграма класів, учасники шаблону, відношення учасників, результати використання, програмна реалізація, відомі застосування.

Тема 7. ПАТЕРНИ ПОВЕДІНКИ

Загальний опис патернів поведінки. Програмні механізми, що використовуються при застосуванні патернів поведінки. Спільність, різниця, способи реалізації. Випадки та приклади застосування. Патерни Ланцюжок Відповідальності (Change of Responsibility), Команда (Command), Ітератор (Iterator), Спостерігач (Observer), Посередник (Mediator), Зберігач (Memento), Стратегія (Strategy), Стан (State), Шаблонний метод (Template Method), Відвідувач (Visitor): призначення, синонімічні назви, мотивація використання, застосовність, переваги та недоліки використання, діаграма класів, учасники шаблону, відношення учасників, результати використання, програмна реалізація, відомі застосування.

Тема 8. ЗАГАЛЬНІ ВИСНОВКИ СТОСОВНО ПАТЕРНІВ ПРОЕК-ТУВАННЯ КЛАСІВ/ОБ'ЄКТІВ (GOF)

Загальні принципи використання патернів проектування класів/об'єктів: породжувальних, структурних, патернів поведінки. Передумови, недоліки та переваги застосування. Використання патернів проектування – як спосіб створення якісного програмного коду. Порівняльна характеристика, способи та приклади сумісного застосування патернів проектування.

Лабораторне – лабораторні заняття відсутні**Курсовий проект/курсова робота/РГР/Контрольна робота:**

1. Загальна характеристика структурних патернів.
2. Породжувальні патерни.
3. Патерни поведінки.

Самостійна робота здобувача:

1. Розв'язання задач із використанням основних принципів програмного забезпечення.
2. Введення в проектування програмного забезпечення із використанням шаблонів. Приклади використання породжувального патерну Фабричний метод (Factory Method).
3. Породжувальний патерн Абстрактна фабрика (Abstract Factory). Розв'язання задач із використанням структурного патерну Адаптер (Adapter).
4. Побудова програмних додатків із використанням структурного патерну фасад (Facade).
5. Загальні принципи побудови програмних додатків із використанням шаблонів поведінки. Патерн ітератор (Iterator).
6. Розробка програмних додатків із використанням патерну поведінки Спостерігач (Observer). Принципи використання патерну поведінки Стратегія (Strategy), програмна реалізація.

17) Іспит: немає.

18) Основна література:

- 1 Brett D.McLanahlin, Gary Pollice, David West. Head first object-oriented analysis and design. – O'Reilly Media, Inc, 2007. –632 p.
- 2 Д. Нильссон. Применение проблемно-ориентированного проектирования и шаблонов проектирования приложений с примера на С# и .NET. – М.: Вильямс, 2008. – 560с.
- 3 Д. Кириевски. Рефакторинг с использованием шаблонов. – М.: Вильямс, 2006. – 400с.
- 4 Э. Фримен, Э. Фримен, К. Сьерра, Б. Гейтс. Паттерны проектирования. – СПб: Питер, 2011. – 656 с.
- 5 Влссидес Джон. Применение шаблонов проектирования.
- 6 Oleksandr Terentyev, Svitlana Tsiutsiura, Tetyana Honcharenko, Tamara Lyashchenko. Multidimensional Space Structure for Adaptable Data Model. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-3, September 2019. P. 7753-7758.
- 7 Kyivska K., Tsiutsiura M., Tsiutsiura S., Terentyev A. Methodology for building project portfolio. Abstracts of II International Scientific and Practical Conference Barcelona, Spain 24-25 February 2020. P. 147-151.
- 8 Kyivska K., Tsiutsiura M., Tsiutsiura S., Terentyev A. Components of information modeling of building objects. Abstracts of I International Scientific and Practical Conference Athens, Greece 29-31 March 2020. P. 138-142.

19) Додаткова література:

- 1 Дополнительные штрихи. – Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 144 с.
- 2 Гамма Э., Хелм Р., Джонсон Р., Влссидес Дж. Приемы объектноориентированного проектирования. Паттерны проектирования. – СПб: Питер, 2001. – 368 с.
- 3 Будаи А. Дизайн-паттерни – простіше простого. – Львів, 2012. – 90с.
- 4 Cooper J.W. Java Design Patterns. Addison-Wesley Professional, 2000 – 542.
- 5 James W. Cooper. Introduction to design patterns in C#. – IBM T J Watson Research Center, 2002. – 424 p.
- 6 Jean Paul V.A. Design Patterns in C#. 2012. – 94 p.
- 7 <http://library.knuba.edu.ua/>
- 8 <http://cpp-reference.ru/patterns/>
- 9 <http://www.dofactory.com/Patterns/Patterns.aspx>
- 10 <http://skillcoding.com/Default.aspx?id=222>
- 11 <http://msdn.microsoft.com/ru-ru/library/dd460654.aspx#Classes>
- 12 <http://www.skillcoding.com/Default.aspx?id=123>

20) Робоче навантаження здобувача, необхідне для досягнення результатів навчання

№	Форма заняття	Кількість годин аудиторні/ СРС
1.	Лекція	-
2.	Практичне заняття	50/25
3.	Лабораторні заняття	-
4.	КП/КР/РГР/Контр.роб.	1 Контрольна робота/12
5.	Форма контролю	залік '6
	Всього годин	50/100
21) Сума всіх годин:		150
22) Загальна кількість кредитів ECTS		5,0
23) Кількість годин (кредитів ECTS) аудиторного навантаження:		50 (1,67)
24) Кількість необхідних годин (кредитів ECTS) СРС для забезпечення аудиторного навантаження:		43 (1,43)
25) Кількість годин (кредитів ECTS) СРС, забезпечених навчальним планом:		100 (3,33)
26) Розробник слабусу: к.т.н., доц. Цюцюра М.І., к.т.н., доц. Київська К.І.		



(дата і місце розробки)

Затверджено:



(місце відповідального кафедролога)