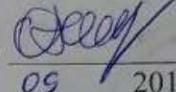


**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ**

Кафедра комп'ютерних технологій і мехатроніки

«ЗАТВЕРДЖУЮ»

Гарант освітньо-професійної програми
«Програмне забезпечення систем»
першого (бакалаврського) рівня вищої
освіти, завідувач кафедри КТМ, д.т.н.,

професор  Ніконов О.Я.
« 03 » 09 2019р.

**СИЛАБУС
ПРОФЕСІЙНА ПРАКТИКА ПРОГРАМНОЇ ІНЖЕНЕРІЇ /
PROFESSIONAL PRACTICE OF SOFTWARE ENGINEERING
SYLLABUS**

освітній ступінь	бакалавр / bachelor
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering
освітня програма	Програмне забезпечення систем / Systems Software

Харків 2019

Автор: Тімонін Володимир Олексійович, кандидат технічних наук,
старший науковий співробітник, доцент

Силабус розглянуто та затверджено на засіданні кафедри комп'ютерних
технологій і мехатроніки, протокол № 18 від «27» червня 2019 р.

СИЛАБУС

ПРОФЕСІЙНА ПРАКТИКА ПРОГРАМНОЇ ІНЖЕНЕРІЇ /

PROFESSIONAL PRACTICE OF SOFTWARE ENGINEERING

SYLLABUS

освітній ступінь	бакалавр / bachelor
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering
освітня програма	Програмне забезпечення систем / Systems Software

ВСТУП

Швидке зростання областей застосування і масштабів програмного забезпечення призвів до переходу від технології індивідуального програмування окремих невеликих програм до колективного створення великих комплексів програм інженерними методами проектування і розробки.

Здобуті теоретичні знання та практичні навички дозволяють фахівцю з розробки програмного забезпечення проводити комплексну розробку програмного забезпечення, пов'язаного з програмуванням і моделюванням систем автоматизованого управління технічними та виробничими процесами з використанням сучасних технологій розробки програмного забезпечення, його супроводження та оновлення на відповідному професійному рівні. Тому вивчення дисципліни «Професійна практика програмної інженерії» є невід'ємною частиною у процесі навчання студентів за спеціальності 121 «Інженерія програмного забезпечення».

Програма вивчення навчальної дисципліни «**Професійна практика програмної інженерії**» складена відповідно до освітньо-кваліфікаційної характеристики та навчального плану підготовки **бакалавра спеціальності 121 «Інженерія програмного забезпечення»**.

Викладання дисципліни розраховано на виконання студентами практичних робіт та самостійних завдань у програмному середовищі MS Visual Studio.

1. МЕТА, ПРЕДМЕТ ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. **Метою** викладання навчальної дисципліни є вивчення та засвоєння методів та засобів професійної практики програмної інженерії в систематизованому вигляді для їх застосування на практиці, отримання практичних навичок ведення професійної діяльності в умовах наближених до реальних.

1.2. **Предметом** вивчення навчальної дисципліни є теорія і практика основ професійної діяльності в галузі розробки програмного забезпечення.

1.3. **Основними завданнями** вивчення навчальної дисципліни є формування у студентів базових теоретичних знань та практичних навичок з професійної практики програмної інженерії у галузі сучасних технологій програмування; ознайомлення з технічними, алгоритмічними, програмними і технологічними рішеннями у області програмування; вироблення практичних навичок аналітичного та експериментального дослідження основних методів і засобів; формування практичних навичок розробки технічної документації, ліцензійних угод та авторських договорів.

1.4. **По завершенні вивчення дисципліни студенти повинні:**

знати:

що таке інженерія програмного забезпечення; основні концепції, що лежать в основі процесу створення ПЗ; принципи інженерії програмного забезпечення; вимоги проектування, конструювання та тестування; основні моделі процесу

створення ПЗ; архітектуру програмного забезпечення та концепції архітектур проектування ПЗ; етапи планування програмного проекту; методи визначення вимог до програмного забезпечення; основні розділи та принципи складання технічного завдання; загальне уявлення про верифікацію та атестацію ПЗ та методи статичної верифікації; прийоми налагодження і ручного тестування ПЗ; відмінні риси системного, навантажувального та граничного тестування інформаційних систем; основні задачі, що стоять перед керівником програмного проекту; засоби та прикладні інтерфейси програмування; клієнт/серверну архітектуру ПЗ; порядок реєстрації авторського права на комп'ютерну програму;

вміти:

визначати вимоги до програмного забезпечення; планувати процес розроблення програмних систем; розробляти технічне завдання на створення програмного продукту; застосовувати сучасні практики програмної інженерії в процесі розроблення програмного забезпечення; використовувати принципи проектування та шаблони при проектуванні; реалізовувати простий графічний користувацький інтерфейс системи; використовувати прості методи вимірювання ПЗ; розробляти документацію на ПЗ, в тому числі документі для реєстрації авторського права на програмний продукт.

1.5. Міждисциплінарні зв'язки:

Вивченню дисципліни:

передують: “Алгоритмізація та програмування”, “Алгоритми і структури даних”, “Об'єктно-орієнтоване програмування”, “Крос-платформне програмування”, “Аналіз вимог до програмного забезпечення”, “Проектування та конструювання ПЗ”, “Стандартизація та сертифікація ПЗ”, “Якість програмного забезпечення та тестування”, “Групова динаміка і комунікації”.

потребують її вивчення: “Переддипломна практика”, “Дипломне проектування”.

2. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Характеристика навчальної дисципліни
Кількість кредитів - 3 Кількість годин - 90	<u>нормативна</u> (нормативна, за вибором ВНЗ, за вибором студента)
Семестр викладання дисципліни	8
Вид контролю:	<u>залік</u> (залік, екзамен)
Розподіл часу:	
- лекції (годин)	8
- практичні, семінарські (годин)	-
- лабораторні роботи (годин)	16
- самостійна робота студентів (годин)	66

- курсовий проект (годин)	-
- курсова робота (годин)	-
розрахунково-графічна робота (контрольна робота)	-
підготовка та складання екзамену	-

3. РОЗПОДІЛ ДИСЦИПЛІНИ У ГОДИНАХ ЗА ФОРМАМИ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ ТА ВИДАМИ НАВЧАЛЬНИХ ЗАНЯТЬ

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оціню- вання (бал)
1	2	3	4
Тема 1. Інженерія вимог.			
Знати основні концепції, що лежать в основі процесу створення ПЗ; основні моделі процесу створення ПЗ; архітектуру програмного забезпечення та концепції архітектур проектування ПЗ; етапи планування програмного проекту; методи проектування ПЗ; основні розділи та принципи складання технічного завдання.	Лекція №1. Процес розробки програмного забезпечення. <i>Основні питання:</i> 1. Основи програмних вимог. 2. Стандарти для технічного завдання. <i>Список рекомендованих джерел:</i> Основний – 1,2,3,6 Додатковий – 8,9,10,11 Інтернет-ресурси –12,13,14,15,16	2	
	Самостійна робота студента (СРС ¹). Основи програмних вимог. <i>Основні питання:</i> 1. Визначення вимог. 2. Функціональні і нефункціональні вимоги. 3. Вимоги з кількісної оцінки. 4. Системні та програмні вимоги. 5. Процес роботи з вимогами. 6. Витяг вимог. 7. Аналіз вимог. 8. Специфікація вимог.	8	
	Практична робота №1 (ПР1). Розробка технічного завдання. <i>Мета роботи - отримати практичні</i>	4	0...100

¹ Завданнями для самостійної роботи є вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем, а також в практичному відпрацюванні незавершених завдань в практичних роботах.

забезпечення; планувати процес розроблення програмних систем; розробляти технічне завдання на створення програмного продукту; застосовувати сучасні практики програмної інженерії в процесі розроблення програмного забезпечення.	навички зі створення технічного завдання. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи. 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Розробка технічної документації для програмного забезпечення. Основні питання: 1. Призначення технічного завдання. 2. Склад технічного завдання. 3. Стандарти для технічного завдання.	8	
Тема 2. Проектування програмного забезпечення			
Знати архітектуру програмного забезпечення та концепції архітектур проектування ПЗ; етапи планування програмного проекту; методи визначення вимог до програмного забезпечення; засоби та прикладні інтерфейси програмування; клієнт/серверну архітектуру ПЗ. Вміти застосовувати сучасні практики програмної інженерії в процесі	Лекція №2. Проектування програмного забезпечення. Основні питання: 1. Загальні концепції проектування. 2. Структура і архітектура програмного забезпечення. 3. Стратегії та методи проектування програмного забезпечення. Список рекомендованих джерел: Основний – 3,4,5,6,7 Додатковий – 8,9,10,11 Інтернет-ресурси –12,13,14,15,16	2	
	СРС. Проектування програмного забезпечення. Основні питання: 1. Загальні концепції проектування. 2. Процес проектування. 3. Техніки застосування. 4. Ключові питання проектування.	8	
	ПР2. Проектування програмного забезпечення. Створення	4	0... 100

розроблення програмного забезпечення; використовувати принципи проектування та шаблони при проектуванні; реалізовувати простий графічний користувацький інтерфейс системи.	призначеного для користувача інтерфейсу. <i>Мета роботи - отримати практичні навички зі створення призначеного для користувача інтерфейсу.</i> План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, розробка призначеного для користувача інтерфейсу). 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Структура і архітектура програмного забезпечення. <i>Основні питання:</i> 1. Структура і архітектура програмного забезпечення. 2. Аналіз якості та оцінка програмного дизайну. 3. Нотації проектування. 4. Стратегії та методи проектування програмного забезпечення.	8	
Тема 3. Конструювання програмного забезпечення			
Знати основи конструювання ПЗ; види розробок ПЗ; методології конструювання ПЗ; механізми управління конструюванням; методи розробки програмних засобів для вирішення практичних завдань; методи розробки компонентів програмних комплексів з використанням	Лекція №3. Конструювання програмного забезпечення. <i>Основні питання:</i> 1. Основи конструювання. 2. Управління конструюванням. <i>Список рекомендованих джерел:</i> Основний – 3,4,5,6,7 Додатковий – 8,9,10,11 Інтернет-ресурси –12,13,14,15,16	2	
	СРС. Основи конструювання. <i>Основні питання:</i> 1. Мінімізація складності. 2. Очікування змін. 3. Конструювання з можливістю перевірки. 4. Стандарти в конструюванні.	8	
	ПРЗ. Конструювання програмного забезпечення. Створення	4	0... 100

сучасних програмних засобів і технологій розробки. Вміти розробляти моделі компонентів інформаційних і автоматизованих систем; сполучати апаратні і програмні засоби в складі автоматизованих і автоматичних систем.	програмного коду програмного продукту. <i>Мета роботи - отримати практичні навички зі створення програмного коду програмного продукту.</i> План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Управління конструюванням. <i>Основні питання:</i> 1. Моделі конструювання. 2. Планування конструювання. 3. Вимірювання в конструюванні. 4. Мови конструювання.	8	
Тема 4. Тестування програмного забезпечення			
Знати прийоми налагодження і ручного тестування ПЗ; відмінні риси системного, навантажувального та граничного тестування інформаційних систем; модель оцінки ступеня тестування програмного продукту; загальне уявлення про верифікацію та атестацію ПЗ та методи статичної верифікації; порядок реєстрації авторського права на комп'ютерну програму.	Лекція №4. Тестування програмного забезпечення. <i>Основні питання:</i> 1. Основи тестування. 2. Рівні тестування. 3. Техніки тестування. 4. Процес тестування. <i>Список рекомендованих джерел:</i> Основний – 3,4,5,6,7 Додатковий – 8,9,10,11 Інтернет-ресурси –12,13,14,15,16	2	
	СРС. Тестування програмного забезпечення. <i>Основні питання:</i> 1. Термінологія тестування. 2. Цілі тестування. 3. Види тестування. 4. Оцінка програм в процесі тестування.	8	
	ПР4. Створення документів на авторське право. <i>Мета роботи - отримати практичні навички зі створення документів на</i>	4	0...100

Вміти побудувати керуючий граф програми для тестування; оцінити складність тестування програмного продукту з використанням математичної моделі; побудувати набір тестів для тестування складної інформаційної системи; розробляти документацію на ПЗ.	<i>авторське право.</i> План заняття: 5. Актуалізація теоретичного матеріалу. 6. Виконання завдань практичної роботи. 7. Презентація і захист виконаної роботи. 8. Тестування знань.		
	СРС. Стандартизація, сертифікація і ліцензування програмного продукту. <i>Основні питання:</i> 1. Документування і стандартизація ПО. 2. Ліцензування ПО. 3. Авторське право на програмний продукт.	10	
Разом		120	0...100

5. КОНТРОЛЬ ТА ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Під час вивчення дисципліни «Професійна практика програмної інженерії» викладачем здійснюється поточний та підсумковий контроль з використанням тестових завдань. Поточний контроль та оцінювання передбачає:

- перевірку рівня засвоєння теоретичного матеріалу (тестування за матеріалами лекції);
- захист практичних робіт (перевірка розроблених проектів та тестування за матеріалами практичних робіт);
- перевірка засвоєння матеріалу, що винесений на самостійне опрацювання під час фронтального опитування на лекції.

Тестування знань студентів здійснюється під час проведення практичних занять за допомогою автоматизованої системи знань. Підсумкова оцінка (ПО) знань з навчальної дисципліни визначається як середньозважена результатів засвоєння окремих блоків в інтервалі (1... 100) балів:

$$ПО = \frac{\sum_{i=1}^I K_i}{N},$$

де K – кількість балів за кожен блок,

I – кількість зданих блоків,

N – загальна кількість блоків за дисципліну.

У разі незгоди студента з підсумковою оцінкою, підсумковий контроль знань та компетентностей студента з навчальної дисципліни здійснюється на підставі заліку.

Відповідність підсумкових семестрових рейтингових оцінок у балах оцінкам за національною шкалою та шкалою ECTS

Оцінка в балах	Оцінка за національною шкалою	Оцінка за шкалою ЄКТС	
		Оцінка	Критерії
90-100	Відмінно	A	“Відмінно” - теоретичний зміст курсу освоєний цілком, без прогалин, необхідні практичні навички роботи з освоєним матеріалом сформовані, усі передбачені програмою навчання навчальні завдання виконані, якість їхнього виконання оцінено числом балів, близьким до максимального.
82 – 89	Добре	B	“Дуже добре” - теоретичний зміст курсу освоєний цілком, без прогалин, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, усі передбачені програмою навчання навчальні завдання виконані, якість виконання більшості з них оцінено числом балів, близьким до максимального.
75 – 81		C	“Добре” - теоретичний зміст курсу освоєний цілком, без прогалин, деякі практичні навички роботи з освоєним матеріалом сформовані недостатньо, усі передбачені програмою навчання навчальні завдання виконані, якість виконання жодного з них не оцінено мінімальним числом балів, деякі види завдань виконані з помилкам
67 – 74	Задовільно	D	“Задовільно” - теоретичний зміст курсу освоєний частково, але прогалини не носять істотного характеру, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, більшість передбачених програмою навчання навчальних завдань виконано, деякі з виконаних завдань, можливо, містять помилки.
60 – 66		E	“Достатньо” - теоретичний зміст курсу освоєний частково, деякі практичні навички роботи не сформовані, багато передбачені програмою навчання навчальні завдання не виконані, або якість виконання деяких з них оцінено числом балів, близьким до мінімального.
35 – 59	Незадовільно	FX	“Незадовільно” - теоретичний зміст курсу освоєний частково, необхідні практичні навички роботи не сформовані, більшість передбачених програм навчання навчальних завдань не виконано, або якість їхнього виконання оцінено числом балів, близьким до мінімального; при додатковій самостійній роботі над матеріалом курсу можливе підвищення якості виконання навчальних завдань(з можливістю повторного складання)

1 – 34		Ф	“Неприйнятно” - теоретичний зміст курсу не освоєно, необхідні практичні навички роботи не сформовані, усі виконані навчальні завдання містять грубі помилки, додаткова самостійна робота над матеріалом курсу не приведе до якого-небудь значимого підвищення якості виконання навчальних завдань. (з обов'язковим повторним курсом)
--------	--	---	---

6. ЧАС ТА МІСЦЕ ПРОВЕДЕННЯ:

- аудиторні заняття – відповідно до розкладу ХНАДУ, ауд. 214, 216;
- позааудиторна робота – самостійна робота студента, результат виконання якої висвітлено засобами Visual Studio.

7. СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний

1. Государственный стандарт СССР «Техническое задание. Требования к содержанию и оформлению». ГОСТ 19.201-78. – М.: Госстандарт СССР, 1978 – 4 с.
2. Государственный стандарт СССР «Информационная технология. Техническое задание на создание автоматизированной системы». ГОСТ 34.602-89. – М.: Госстандарт СССР, 1989 – 11 с.
3. Лаврищева К.М. Програмна інженерія. – К.:НАНУ, 2008.–319 с.
4. Лаврищева Е.М. , Петрухин В.А. Методы и средства инженерии программного обеспечения. – М.: Московский физико-технический институт, 2006. – 304 с.: ил.
5. Орлов С.А. Программная инженерия. Технологии разработки программного обеспечения. 5-е издание. – СПб.: Питер, 2017. – 640 с.: ил.
6. Саммервил И. Инженерия программного обеспечения.- М.: Вильямс, 2002.- 620 с.
7. Тімонін В.О. Методичні вказівки до проведення практичних робіт з дисципліни «Професійна практика програмної інженерії» для студентів за спеціальністю 121 «Інженерія програмного забезпечення». – Харків: ХНАДУ, 2019. – 198 с. (в електронній формі).

Додатковий

8. Батоврин В.К. Толковый словарь по системной и программной инженерии. — М.: ДМК Пресс, 2012. — 280 с.: ил.
- 9.Бахтизин В.В. Технология разработки программного обеспечения : учеб. пособие / В.В. Бахтизин, Л.А. Глухова. – Минск: БГУИР, 2010. – 267 с.: ил.
10. Буч Г. Объектно-ориентированное проектирование с примерами применения. Пер. с англ. – М.: Конкорд, 1992, – 406 с.: ил.
11. Рекомендации по преподаванию программной инженерии и информатики в университетах = Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering; Computing

Curricula 2001: Computer Science: пер. с англ. — М.: ИНТУИТ.РУ «Интернет-Университет Информационных Технологий», 2007. — 462 с. : ил.

Интернет-ресурси

12. <https://www.computer.org/web/swebok>
13. <https://msdn.microsoft.com>.
14. <https://uk.wikipedia.org>.
15. <https://studfiles.net>
16. <https://cs.fit.edu/>