Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Кафедра системного проектування

“**ЗАТВЕРДЖУЮ**”

Декан факультету електроніки та

комп'ютерних технологій

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Фургала Ю.М.

“\_\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 року

**УПРАВЛІННЯ ІТ ПРОЕКТАМИ (інженерія програмного забезпечення)**

**ПРОГРАМА**

**навчальної дисципліни**

**підготовки бакалаврів**

**галузі знань 12 Інформаційні технології**

**спеціальності 126 Інформаційні системи та технології**

**факультету електроніки та комп’ютерних технологій**

**2020** рік

РОЗРОБЛЕНО ТА ВНЕСЕНО:

Львівським національним університетом імені Івана Франка

РОЗРОБНИКИ ПРОГРАМИ:

Ненчук Т.М., канд. фіз.-мат. наук, доцент кафедри системного проектування факультету електроніки та комп’ютерних технологій

Затверджено на засіданні Вченої ради факультету електроніки та комп'ютерних технологій

“\_31\_” \_\_серпня\_\_ 2020 року, протокол №\_\_\_

Голова Вченої ради Фургала Ю.М.

Обговорено та рекомендовано до затвердження Навчально-методичною радою факультету електроніки та комп'ютерних технологій

“\_31\_” \_\_\_серпня\_\_\_ 2020 року, протокол №\_\_

Голова Навчально-методичної ради

факультету електроніки та комп'ютерних технологій Лучечко А.П..

Програму схвалено на засіданні кафедри системногопроектування

“\_\_31\_\_”\_\_серпня\_\_\_2020 року, протокол № \_\_\_

Завідувач кафедри

системного проектування Шувар Р.Я.

© Ненчук Т.М. 2020 рік

# Загальні Відомості

Програма вивчення навчальної дисципліни «**Управління ІТ проектами (інженерія програмного забезпечення)**» складена відповідно до освітньо-професійної програми підготовки здобувачів першого (бакалаврського) рівня вищої освіти спеціальності **126- Інформаційні системи та технології** і є нормативною дисципліною циклу професійно-орієнтованих дисциплін.

**Міждисциплінарні зв'язки:** навчальна дисципліна «**Управління ІТ проектами (інженерія програмного забезпечення)**» вивчається у логічному взаємозв'язку з іншими навчальними дисциплінами, що обумовлює необхідність постійного обліку та реалізації викладачами існуючих міжпредметних зв'язків з суміжними навчальними курсами, зокрема, такими як "Програмування", "Об'єктно-орієнтоване програмування", "Алгоритми та структури даних" та інші. Знання отримані під час вивчення дисципліни будуть використовуватися при вивченні курсів “Управління ІТ проектами (менеджмент)”, "Сучасні парадигми програмування" та під час написання курсової роботи.

**Предметом** вивчення навчальної дисципліни є методології сучасних методів управління проектами, в тому числі, їх життєвий цикл, методи командної роботи в проекті, методи аналізу та верифікації вимог замовників та команди до проектів, оцінки часових рамок проектів, можливості групової динаміки та соціальних комунікацій, зокрема, презентування ідей, результатів командної роботи необхідних для роботи над програмними проектами.

Програма навчальної дисципліни складається з двох модулів та чотирьох змістових модулів.

**1. Мета та завдання навчальної дисципліни**

 **1.1. Метою викладання навчальної дисципліни** є набуття теоретичних знань з управління проектами та засвоєння практичних навичок їх використання, зокрема, отримання студентами основ знань щодо управління ІТ проектами, зокрема, ознайомлення з їх життєвим циклом, методами командної роботи в проекті, методами аналізу та верифікації вимог замовників та команди до проектів, оцінками часових рамок проектів, можливостями групової динаміки та соціальних комунікацій, зокрема, презентування ідей, результатів командної роботи необхідних для роботи над програмними проектами.

**1.2. Основні завдання вивчення дисципліни:** У результаті вивчення курсу студенти ознайомлюються із широким набором технологій у галузі управління ІТ проектами в області, яка охоплює інженерію програмного забезпечення та командну роботу.

**1.3. Згідно з вимогами освітньо-професійної програми студенти повинні:**

**знати:** набір методів та інструментів, які дозволяють ефективно налаштовувати, запускати, оцінювати і постійно покращувати командну роботу над проектом;

основні поняття інженерії програмного забезпечення, зокрема, системи контролю версій, правила написання коду, методи функціонального та об'єктно - орієнтованого аналізу та реалізації ІТ проектів;

**вміти**: системно мислити та застосовувати творчі здібності до опрацювання вимог до проектів, формування принципово нових ідей, ефективно працювати як індивідуально, так і у складі команди, оцінювати отримані результати, представляти результати роботи команди проекту та аргументовано захищати прийняті рішення.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Форма* *навчання* | *Семестр* | *Всього**кредитів/годин* | *Розподіл навчального часу за видами занять* [[1]](#footnote-1) | *Семестрова атестація* |
| *Лекції* | *Практичні заняття* | *Семінарськізаняття* | *Лабораторні роботи* | *СРС* |
| *Денна* | 2 | 5/150 | 32 | - | - | 32 | 86 | залік |

**ІІ. Інформаційний обсяг навчальної дисципліни**

***Змістовий модуль 1*. Основні поняття управління проектом. Команда проекту.**

**Тема 1. Основні поняття та методологія управління ІТ проектами.**

**Тема 2. Командна робота. Cтруктура і завдання проектної команди.**

***Змістовий модуль 2*. Командна робота з кодом.**

**Тема 3.** **Системи контролю версій.**

**Тема 4. Правила написання коду в команді.**

***Змістовий модуль 3*. Виявлення, аналіз, специфікація і верифікація вимог до ІТ проектів.**

**Тема 5. Вимоги до проектів.**

**Тема 6. Моделювання предметної області проектів.**

**Тема 7. UML опис предметної області проектів, діаграми, генерація коду. Rational Unified Process.**

**Тема 8. Оцінка тривалості завдань у проектах**

***Змістовий модуль 4*. Презентація ІТ проекту.**

**Тема 9. Підготовка до та організація представлення ідей та результатів проектів.**

# ІІІ. ПРИБЛИЗНИЙ ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

1. Типи ролей в групі. Робота в команді та управління конфліктами. Формати командної роботи.
2. Створення проекту в Github, додавання коду в проект та розв’язання конфліктів.
3. Розробка вимог до проекту. Збір вимог до проекту, їх пріоритизація та представлення у вигляді користувацьких історій.
4. Проектування розробки. Оцінка тривалості задач та методики оцінювання.
5. Реалізація проекту. Проведення перегляду коду командою. Методики парного перегляду коду та розробки.
6. Тестування розробки проекту.
7. Підготовка та створення ефективної презентації проекту.
8. Захист проекту командою.

**ІV. Рекомендована література**

# Методичні вказівки

1. Ненчук Т.М. Методичні рекомендації до виконання лабораторних робіт з курсу «Управління ІТ проектами (інженерія програмного забезпечення)». Електронна версія.

***Базова література***

1. Pro Git, 2009. – 288 с.
2. Ben Lynn. Магія Git. – 54 с.
3. Мартин Р. К. Чистый код: создание, анализ и рефакторинг. Библиотека программиста / Роберт К. Мартин., 2016. – 464 с.
4. Пассиг К., Яндер Й. Программирование без дураков. – 2016. - 416 с.
5. Макконнелл С. Сколько стоит программный проект / Стив Макконнелл., 2007. – 304 с.
6. Мельник О.Г., Шпак Ю.Н. Декомпозитна модель альтернатив формування команд для ІТ компаній. – Technology Audit and Production Reserves. - №3/5(23), 2015, с. 11-15.
7. Martin Olson. Foundations of the scaled Agile Frameworks. Be Agile. Scale Up. Stay Lean. – 2014. – Scaled Agile, Inc.
8. Иванов Д. Ю., Новиков Ф. А. Основы моделирования на UML: Учеб. пособие. – СПб.: Изд-во Политехн. ун-та, 2010. – 249 с.
9. Лаврищева Е. М., Петрухин В. А. Методы и средства инженерии программного обеспечения.- 2007. - 415 с.
10. Соммервилл И. Инженерия программного обеспечения. - 2002. – 624 с.

***Допоміжна література***

1. A Guide to the Project Management Body of Knowledge (PMBOK Guide) – Fourth Edition. – Project Management Institute, USA.

# *Інформаційні ресурси*

1. Базове використання Git. [Електронний ресурс]. – 2009. – Режим доступу до ресурсу: uk.enetri.com/2009/10/26/2.html
2. Управління проектами. Підручник. [Електронний ресурс]. – 2017.- Режим доступу до ресурсу: buklib.net/books/21936
3. Лозовицький Д., Бачинський Т. Особливості функціонування самокерованих ІТ команд. [Електронний ресурс]. – 2017. - Режим доступу до ресурсу: aphd.ua/publication-145
4. Star UML. Руководство пользователя [Електронний ресурс]. – 2015.- Режим доступу до ресурсу: staruml.sourceforge.net/docs/user-guide(ru)/user-guide.pdf
5. Стилі програмування [Електронний ресурс]. – 2016. - Режим доступу до ресурсу: lib.mdpu.org.ua/e-books/
6. Коментарі [Електронний ресурс]. – 2015. - Режим доступу до ресурсу: snowdream.github.io/javascript-style-guide/
7. How to Determine Your Software Application Size Using Function Point Analysis. [Електронний ресурс]. – 2017. - Режим доступу до ресурсу: <https://alvinalexander.com/FunctionPoints/>
8. Data flow [Електронний ресурс]. – 2012. - Режим доступу до ресурсу: <https://www.visual-paradigm.com/tutorials/data-flow-diagra>.
9. Git Workflow | Atlassian Git Tutorial [Електронний ресурс]. – 2017. - Режим доступу до ресурсу: <https://www.atlassian.com/git/tutorials/comparing-workflows>
10. Comparing Logical and Physical ERD [Електронний ресурс]. – 2017. - Режим доступу до ресурсу: https://www.visual-paradigm.com/tutorials/compare-logical
1. [↑](#footnote-ref-1)