

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «КРОС-ПЛАТФОРМНЕ ПРОГРАМУВАННЯ»



Ступінь освіти	бакалавр
Спеціальність	122 Комп'ютерні науки
Освітня програма	Комп'ютерні науки
Загальний обсяг	4 кредити ЄКТС (120 годин)
Тривалість викладання	6 семестр (11, 12 чверті)
Заняття:	
лекції:	2 години на тиждень
лабораторні заняття:	2 години на тиждень
Мова викладання	українська

Сторінка курсу в СДО НТУ «ДП»:

Кафедра, що викладає Програмного забезпечення комп'ютерних систем



Викладач:

Лактіонов Іван Сергійович

доктор технічних наук, доцент, професор кафедри програмного забезпечення комп'ютерних систем

Персональна сторінка

<https://pzks.nmu.org.ua/ua/teachers/laktionovis.php>

E-mail: laktionov.i.s@nmu.one

1. Анотація до курсу

Крос-платформне програмування є комплексом знань та практичних навичок, які направлені на вдосконалення розробки програмного забезпечення із використанням інструментів для гнучкого перенесення розробок між різноманітними платформами. У програмі цієї дисципліни приділяється увага технології та практичним навичкам розробки прикладного програмного забезпечення під різні операційні системи з використанням прогресивного фреймворку Vue.js, бібліотеки wxWidgets та фреймворку Qt. Під час навчання виконується розробка навчальних прикладних веб-додатків, прикладних програм для операційних систем Android, Windows, iOS.

Сучасні веб-додатки побудовані на основі багаторівневої архітектури, включають в себе динамічне створення та інтерпретацію коду, а також реалізацію конкретного режиму взаємодії на основі навігаційної структури проекту. В рамках курсу розглядаються такі питання: структура інтегрованої кібер-фізичної системи підприємства; основні поняття при проектуванні крос-платформних систем; введення до бібліотеки wxWidgets; введення до бібліотеки Qt; Web-sockets; Основи Vue.js: Data binding, Directives, Events, Filters, Computed properties, Rendering Lists, Components, Lifecycle hooks; Основи Quasar: Quasar CLI, Library of Components, Plugins, Quasar Directives, Platform Detection, Utilities; Single page application.

2. Мета та завдання курсу

Мета дисципліни – формування компетентностей щодо вміння розробки прикладного програмного забезпечення під різні операційні системи/платформи.

Завданнями дисципліни є:

- опанування здобувачами освіти основних підходів щодо методів технології крос-платформного програмування;
- формування у здобувачів комплексу знань та практичних навичок із проектування компонентів систем крос-платформного програмування;
- надання здобувачам освіти сучасних практик та навичок роботи з інструментальними засобами при розробці навчальних крос-платформних прикладних програм;
- опанування здобувачами вмісту бібліотеки wxWidgets та можливостями фреймворку Qt.

3. Дисциплінарні результати навчання

Дисциплінарні результати навчання сформовано на основі ПРН освітньо-професійної програми «Комп'ютерні науки» першого (бакалаврського) рівня вищої освіти (ПР9, ПР13).

- володіти навичками використання крос-платформних бібліотек, фреймворків та технологій для побудови програм та бібліотек, які функціонують на різноманітних програмних та апаратних платформах;
- розробляти програмні моделі предметних явищ з застосуванням технологій крос-платформного програмування.

4. Структура курсу

Види та тематика навчальних занять	Внесок в загальну оцінку, %
ЛЕКЦІЇ	40
1. Введення до бібліотеки wxWidgets 1.1. Введення до крос-платформного програмування. Основні поняття та визначення. Класифікація програмного інструментарію для крос-платформного програмування	

Види та тематика навчальних занять	Внесок в загальну оцінку, %
1.2. Особливості встановлення та налагодження бібліотек wxWidgets та Qt 1.3. Внутрішня організація бібліотеки wxWidgets. Структура програми с графічним інтерфейсом користувача на базі класів бібліотеки wxWidgets 1.4. Класи wxWidgets для роботи с елементами графічного інтерфейсу користувача	
2. Введення до бібліотеки Qt 2.1. Склад бібліотеки Qt. Схема створення програм на Qt. 2.2. Механізм сигналів та слотів Qt. Глобальні визначення Qt. Основні класи програм. 2.3. Класи QWidget, QPushButton, QLabel, QLineEdit 2.4. Робота з візуальними елементами управління Qt	
<i>Тестова контрольна робота №1 (за темами 1-2).</i>	20
3. Структура інтегрованої кіберфізичної системи підприємства	
4. Проектуванні крос-платформних систем	
5. Проектуванні крос-платформних систем на базі фреймворку Vue.js	
6. Проектуванні крос-платформних систем на базі фреймворку Quasar.	
7. Проектування Single page application, Progressive web application, Server-side rendering.	
<i>Тестова контрольна робота №2 (за темами 3-7).</i>	20
ЛАБОРАТОРНІ ЗАНЯТТЯ	60
Лабораторна робота 1 Розробка програми обробки файлів даних на базі бібліотеки wxWidgets	
<i>Звіт з роботи № 1 та захист лабораторної роботи.</i>	12
Лабораторна робота 2 Розробка крос-платформної програми на базі бібліотеки Qt.	
<i>Звіт з роботи № 2 та захист лабораторної роботи.</i>	12
Лабораторна робота 3 Проектування запитів за допомогою API.	
<i>Звіт з роботи № 3 та захист лабораторної роботи.</i>	12
Лабораторна робота 4 Проектування прикладної програми Web додатку	
<i>Звіт з роботи № 4 та захист лабораторної роботи.</i>	12
Лабораторна робота 5 Проектування прикладних програм для операційних систем Android, Windows, iOS.	
<i>Звіт з роботи № 5 та захист лабораторної роботи.</i>	12
РАЗОМ	100

5. Технічне обладнання та/або програмне забезпечення

Використовуються лабораторії кафедри програмного забезпечення комп'ютерних систем (комп'ютерне та мультимедійне обладнання). Дистанційна платформа Microsoft Teams, MS Office 365, Moodle, Vue.js, Quasar, Node.js, Visual Studio Code, GIT, бібліотека wxWidgets, фреймворк Qt.

6. Система оцінювання та вимоги

6.1. Навчальні досягнення здобувачів вищої освіти за результатами вивчення курсу оцінюватимуться за шкалою, що наведена нижче:

Рейтингова шкала	Інституційна шкала
90 – 100	відмінно
74-89	добре
60-73	задовільно
0-59	незадовільно

6.2. Здобувачі вищої освіти можуть отримати **підсумкову оцінку** з навчальної дисципліни **на підставі поточного оцінювання знань** за умови, якщо набрана кількість балів з поточного тестування та виконання і захисту лабораторних робіт складатиме не менше 60 балів.

Теоретична частина оцінюється за результатами здачі двох контрольних тестових робіт, кожна з яких містить тестові закриті запитання з однією вірною відповіддю (максимальна кількість – 20 балів за кожною тестовою роботою). Загалом за дві контрольні тестові роботи отримується **максимум 40 балів**, тобто 40% від оцінки за дисципліну.

Лабораторні роботи (5 робіт – у вигляді індивідуального завдання з кожної, розподіл % див. в таблиці розділу 4) виконуються у письмовому вигляді (звіт з кожної роботи оцінюється в межах балів, представлених в таблиці розділу 4, загалом лабораторні враховуються як 60% (максимум 60 балів). При несвоєчасному здаванні роботи оцінка знижується вдвічі. Лабораторні роботи захищаються у вигляді опитування за звітом, і захист враховується, як 50% від оцінки за роботу. У сумі за лабораторну частину курсу при поточному оцінюванні отримується **максимум 60 балів**.

Отримані бали за теоретичну частину та лабораторні роботи додаються і є поточною успішністю за вивчення навчальної дисципліни. Максимально за поточною успішністю здобувач вищої освіти може набрати 100 балів.

Максимальне оцінювання поточного контролю в балах:

Теоретична частина	Практична частина	Разом
40	60	100

6.3. Критерії оцінювання підсумкової роботи. У випадку якщо здобувач вищої освіти за поточною успішністю отримав менше 60 балів та/або прагне поліпшити оцінку, проводиться **підсумкове оцінювання (екзамен)** під час сесії.

Екзамен проводиться у вигляді комплексної контрольної роботи, яка включає запитання з теоретичної та практичної частини курсу. Білет складається з **40 тестових завдань** з чотирма варіантами відповідей, одна правильна відповідь оцінюється в 2 бали (**разом 80 балів**) та **2 завдання** з практичної частини, кожне з запитань оцінюється максимум у 10 балів (**разом 20 балів**). Отримані бали за тестові завдання та завдання з практичної частини

додаються і є підсумковою оцінкою за вивчення навчальної дисципліни. Максимально за підсумковою роботою здобувач вищої освіти може набрати 100 балів.

7. Політика курсу

7.1. Політика щодо академічної доброчесності. Академічна доброчесність здобувачів вищої освіти є важливою умовою для опанування результатами навчання за дисципліною і отримання задовільної оцінки з поточного та підсумкового контролів. Академічна доброчесність базується на засудженні практик списування (виконання письмових робіт із залученням зовнішніх джерел інформації, крім дозволених для використання), плагіату (відтворення опублікованих текстів інших авторів без зазначення авторства), фабрикації (вигадування даних чи фактів, що використовуються в освітньому процесі). Політика щодо академічної доброчесності регламентується положенням "Положення про систему запобігання та виявлення плагіату у Національному технічному університеті "Дніпровська політехніка" <https://cutt.ly/MCfh5kv>

У разі порушення здобувачем вищої освіти академічної доброчесності (списування, плагіат, фабрикація), робота оцінюється незадовільно та має бути виконана повторно. При цьому викладач залишає за собою право змінити тему завдання.

7.2. Комунікаційна політика. Здобувачі вищої освіти повинні мати активовану корпоративну університетську пошту. Усі письмові запитання до викладачів стосовно курсу мають надсилатися на університетську електронну пошту.

7.3. Політика щодо перескладання. Роботи, які здаються із порушенням термінів без поважних причин оцінюються на нижчу оцінку. Перескладання підсумкового оцінювання відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

7.4 Політика щодо оскарження оцінювання. Якщо здобувач вищої освіти не згоден з оцінюванням його знань він може опротестувати виставлену викладачем оцінку у встановленому порядку.

8. Рекомендовані джерела інформації

1. Ровінський В.А. Навчальний посібник з курсу «Кросплатформне програмування». Івано-Франківськ: Прикарпатський національний університет імені Василя Стефаника, 2020. 151с.

2. Крос-платформне програмування: лабораторний практикум з виконання лабораторних робіт / укл. Ю.О.Литвин, Ю.О.Сіциліцин, О.В.Строкань – Мелітополь: Вид-во ТДАТУ, 2019 – 220 с.

3. Heitor Ramon Ribeiro. Vue.js 3 Cookbook: Discover actionable solutions for building modern web apps with the latest Vue features and TypeScript. – Packt Publishing, 2020. – 562 p.

4. Lachlan Miller. Design patterns for vue.js – A test driven Approach to maintainable applications. – 2021. – 163 p.

5. John Au-Yeung. Vue.js 3 by example: Blueprints to learn Vue web development, full-stack development, and cross-platform development quickly. – Pact Publishing. – 2021.– 320 p.

6. Cross-Platform GUI Programming with wxWidgets / Julian Smart, Kevin Hock, Stefan Csomor.: Prentice Hall, 2005. – 744 c.