

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»

Кафедра системного аналізу та обчислювальної математики

(найменування кафедри)

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Розробка графічних інтерфейсів для програмних додатків

(назва навчальної дисципліни)

Освітня програма: Інтелектуальні технології та прийняття рішень в складних системах

(назва освітньої програми)

Спеціальність: 124 – Системний аналіз

(найменування спеціальності)

Галузь знань: 12 – Інформаційні технології

(найменування галузі знань)

Ступінь вищої освіти: бакалаврський

(назва ступеня вищої освіти)

Затверджено на засіданні кафедри

СА та ОМ

(найменування кафедри)

Протокол №18 від 16.08.2021 р.

м. Запоріжжя 2021

1. Загальна інформація	
Назва дисципліни	<i>Розробка графічних інтерфейсів для програмних додатків</i>
Рівень вищої освіти	<i>перший (бакалаврський) рівень</i>
Викладач	<i>Рябенко Антон Євгенович, к.ф.-м.н., доцент</i>
Контактна інформація викладача	<i>Телефон кафедри 769-8247, Телефон викладача +380680689709, rjabenkoae@gmail.com</i>
Час і місце проведення навчальної дисципліни	<i>Предметна аудиторія кафедри, комп'ютерні класи згідно розкладу занять</i>
Обсяг дисципліни	<i>розподіл годин: лекції-14, лабораторні заняття-14, вид контролю- залік,</i>
Консультації	<i>Згідно з графіком консультацій</i>
2. Пререквізити і постреквізити навчальної дисципліни	
<i>Пререквізити: програмування та алгоритмічні мови, сучасні засоби створення та просування інтернет ресурсів, Python для аналізу даних</i>	
3. Характеристика навчальної дисципліни	
Інтегральна компетентність Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми системного аналізу у професійній діяльності або в процесі навчання, що передбачають застосування теоретичних положень та методів системного аналізу та інформаційних технологій і характеризуються комплексністю та невизначеністю умов Загальні компетентності: ЗК01. Здатність до абстрактного мислення, аналізу та синтезу; ЗК04. Знання та розуміння предметної області та розуміння професійної діяльності; ЗК05. Здатність розробляти проєкти та управляти ними.; Фахові компетентності: СК4-Здатність оцінювати ризики, розробляти алгоритми управління ризиками в складних системах різної природи. СК7-Здатність управляти робочими процесами у сфері інформаційних технологій, які є складними, непередбачуваними та потребують нових стратегічних підходів. СК8-Здатність розробляти і реалізовувати наукові та прикладні проєкти в галузі інформаційних технологій та дотичні до неї міждисциплінарні проєкти. СК9-Здатність здійснювати захист прав інтелектуальної власності, комерціалізацію результатів досліджень та інновацій. СК10-Здатність до самоосвіти та професійного розвитку. СК11-Здатність критично осмислювати проблеми у галузі інформаційних технологій і на межі галузей знань та розв'язувати складні задачі у широких або мультидисциплінарних контекстах.	

Результати навчання:

РН3-Застосовувати методи розкриття невизначеностей в задачах системного аналізу, розкривати ситуаційні невизначеності та невизначеності в задачах взаємодії, протидії та конфлікту стратегій, знаходити компроміс при розкритті концептуальної невизначеності.

РН5-Використовувати міри оцінювання ризиків та застосовувати їх при аналізі багатофакторних ризиків в складних системах.

РН 12-Розробляти і реалізовувати наукові і прикладні проєкти у сфері інформаційних технологій, а також дотичні до неї міждисциплінарні проєкти з урахуванням цілей, обмежень, технічних, соціальних, економічних, правових та інших аспектів.

Результати навчання:

В результаті вивчення дисципліни студент повинен

знати:

- основні принципи створення GUI засобами мови Python;
- структуру та властивості графічних компонентів та їх взаємодії;

вміти:

- працювати у середі PyCharm
- створювати GUI за допомогою засобів фреймворку Tkinter

4. Мета вивчення навчальної дисципліни

формування у майбутніх фахівців системи спеціальних знань у сфері розробка GUI програмних додатків з використанням мови програмування Python

5. Завдання вивчення дисципліни

Отримання знань по використанню спеціалізованих бібліотек Python для створення графічних інтерфейсів користувача

6. Зміст навчальної дисципліни

У навчальній дисципліні на практичних прикладах розглядаються принципи створення та використання графічних інтерфейсів програм різного типу

7. План вивчення навчальної дисципліни

№ теми	Назва теми	Форми організації навчання	Кількість годин
1.	Тема 1. Історія та класифікація графічних інтерфейсів користувача (далі - GUI).	Лекції, лабораторні заняття, самостійна робота	4

	Порівняння інтерфейсів командного рядка, віконного інтерфейсу та інтерфейсу користувача. Переваги та недоліки використання GUI. Концепція ергономічності (юзабіліті). Огляд типових екранних форм і стандартних елементів інтерфейсу..		
2	Тема 2. Огляд фреймворків мови програмування Python для створення GUI: PyGUI, wxPython, PyQt5, Tkinter. Основи використання фреймворку Tkinter. Головний цикл подій. Створення кореневого вікна програми. Графічні компоненти (Label, Text, Button, Entry, Radiobutton, Checkbutton, Listbox, Menu). Способи розташування компонентів у кореновому вікні (методи pack та grid). Методи обробки подій та взаємодія компонентів. Змінення властивостей компонентів.	Лекції, лабораторні заняття, самостійна робота	8
3	Тема 3. Графіка в Tkinter та turtle. Створення графічних зображень з модулем turtle. Компонент Canvas та його властивості. Створення графічних об'єктів. Створення анімації. Керування графічними об'єктами.	Лекції, лабораторні заняття, самостійна робота	8
4	Тема 4. Створення інтерфейсів додатків з аналітики даних. Інтеграція GUI з аналітичними модулями. Використання даних numpy та pandas.	Лекції, лабораторні заняття, самостійна робота	8

	Вбудовування графіків та діаграм бібліотек matplotlib та seaborn. Завантаження даних для аналітики з інтернету та файлової системи. Додавання інтерактивності GUI.		
8. Самостійна робота			
<i>Необхідним елементом успішного засвоєння навчального матеріалу дисципліни є самостійна робота студентів з вітчизняною та закордонною спеціальною літературою. Самостійна робота є основним засобом оволодіння навчальним матеріалом у час, вільний від обов'язкових навчальних занять.</i> <i>Основні види самостійної роботи, запропоновані студентам:</i> - <i>вивчення лекційного матеріалу;</i> - <i>робота з рекомендованою літературою;</i> - <i>вивчення окремих тем або питань, що передбачені для самостійного опрацювання;</i> - <i>вивчення основних термінів та понять;</i> - <i>підготовка до лабораторних занять;</i> - <i>підготовка до проміжного та підсумкового контролю;</i> - <i>контрольна перевірка кожним студентом особистих знань за питаннями для самостійного поглибленого вивчення та самоконтролю;</i> - <i>робота над індивідуальним завданням.</i>			
9. Система та критерії оцінювання курсу			
<i>Проміжний контроль знань студентів включає такі заходи:</i> - <i>виконання та захист лабораторних робіт, що передбачені учбовим планом;</i> - <i>аудиторні контрольні роботи (2 за семестр), що проводяться перед кожним підсумковим контролем.</i> <i>Підсумковий контроль знань студентів включає: іспит.</i> <i>Оцінювання знань при проміжному контролі можливо таким чином:</i> - <i>за відмінний захист лабораторних робіт – 50 балів;</i> - <i>при відмінному виконанні всіх завдань підсумкової контрольної роботи (контрольної роботи для заочної форми) – 50 балів.</i>			
10. Політика курсу			
<i>Академічна доброчесність. Студенти не видають за свої результати роботи інших людей. При використанні чужих ідей і тверджень у власних роботах обов'язково</i>			

посилаються на використанні джерела інформації. Під час оцінювання результатів навчання не користуються недозволеними засобами, самостійно виконують навчальні завдання, завдання поточного та підсумкового контролю.

Відвідування занять. Очікується, що всі студенти відвідають усі лекції та практичні заняття курсу. За згоди декана та викладача дозволяється перейти на індивідуальний графік занять. У будь-якому випадку студенти зобов'язані дотримуватися термінів виконання усіх видів робіт, передбачених робочою програмою курсу.

Література

1. Гэддис Т. Начинаем программировать на Python. – 4-е вид. – СПб.: BHV, 2019. – 768 с.
2. Alan D. Moore. Python GUI Programming with Tkinter - Packt Publishing. 2018. 452 pages.
3. Любанович Б. Простой Python. Современный стиль программирования. – 2-е вид. — СПб.: Питер, 2021. — 592 с.
4. Седер Н. Python. Экспресс-курс. – 3-е вид. — Переклад с англ. Е. Матвеев. — Manning Publications Co.; СПб.: Питер, 2019. — 479 с.: ил.
5. Маккинни У. Python и анализ данных. – М.: ДМК Пресс, 2015. — 486 с.
6. Свейгарт Э. Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих – М.: Вильямс, 2017. — 573 с.
7. Элбон К. Машинное обучение с использованием Python. Сборник рецептов – Пер. с англ. — СПб: БХВ-Петербург, 2019. — 384 с.: ил.
8. VanderPlas J. Python Data Science Handbook. — <https://jakevdp.github.io/PythonDataScienceHandbook/>.

Публікації з тематики курсу

немає