Код та назва дисципліни українською мовою/ Назва дисципліни англійською мовою	2-ф09-3 Аналіз та верифікація інтерфейсів користувача комп'ютерних систем/ Analysis and Verification of Computer System User Interfaces
Рекомендується для галузі знань	Для спеціальностей галузі F, G
Кафедра	Електронних обчислювальних машин
П.І.П. НПП *(за можливості)*	доцент, к.т.н. Литвинов Олександр Анатолійович доцент к.ф.-м. н. Карпенко Надія Валеріївна
Рівень ВО	Другий (магістерський)
Курс, семестр *(в якому буде викладатись)*	Курс: 1 курс 2 семестр
Мова викладання	Українська
Пререквізити (передумови вивчення дисципліни)	Поняття про проєктування інтерфейсу користувача
Чому це цікаво/треба вивчати	У контексті розробки комп'ютерних систем поняття інтерфейсу виходить за межі створення візуальних оболонок для сайтів чи мобільних додатків. Інтерфейс — це фундаментальна, яка розділяє рівні системи та визначає правила взаємодії між ними. Цей оригінальний авторський курс пропонує цілісний, системний підхід до проєктування інтерфейсів, з вивченням спільних законів, архітектурних паттернів та методів верифікації, які є універсальними для трьох китів сучасних обчислювальних систем: User Interfaces, Software Interfaces, Hardware Interfaces. Студенти зможуть побачити спільну математичну та логічну природу там, де зазвичай бачать різні технології. Наприклад, як закон когнітивної психології Міллера (обмеження пам'яті людини) однаково впливає і на архітектуру складного графічного меню, і на декомпозицію ендпоінтів у розподіленому API. В інженерних сферах, де ціна помилки є критичною (авіоніка, медицина, промислова автоматизація), правильне проєктування та математична верифікація інтерфейсу — це питання живучості всієї системи. Курс дає відповіді, як проектувати стійкі контракти та інтерфейси, що захищені від збоїв, здатні працювати в Real-time режимі та витримувати екстремальні навантаження.

Перелік тем з дисциплін	✓ Вступ до інженерії інтерфейсів та людино-комп'ютерної взаємодії (HCI). ✓ Користувальницький інтерфейс. ✓ Математичні моделі та закони пов'язані з інтерфейсами. ✓ Принципи та архітектурні паттерни побудови інтерфейсів. ✓ Продуктивність, асинхронність та чуйність інтерфейсів (Responsiveness). ✓ Верифікація та тестування інтерфейсів: від контрактів до UX. ✓ Проєктування для критичних систем (Safety-Critical UI). ✓ Сучасні тренди, кросплатформеність та майбутнє інтерфейсів.
Як можна користуватися набутими знаннями і уміннями *(компетентності)*	Набуті компетенції дозволяють працювати на різних рівнях розробки та супроводу комп'ютерних систем: • На користувальницькому рівні: Usability, UI/UX, використання паттернів (MVC, MVP, MVVM) та компонентно-орієнтованої архітектури. Тестування інтерфейсу. • На системно-архітектурному рівні: Вміння будувати масштабовані системи на основі інтерфейсних контрактів, ізолювати рівні системи за допомогою патернів та керувати складними станами даних. • На програмно-сервісному рівні: Проєктування стійких, зворотно-сумісних системних і прикладних API; впровадження автоматизованого контрактного тестування. • На апаратно-цифровому рівні: розуміння та створення інтерфейсів для вбудованих та IoT-систем з урахуванням обмежених обчислювальних ресурсів, енергоефективності та специфіки передачі даних у реальному часі. • На рівні системної безпеки: Проєктування відмовостійких інтерфейсів для диспетчерських пультів, медичного чи промислового обладнання з мінімальним ризиком «людського фактору». Створення інтерфейсів для критичних систем із впровадженням функцій захисту від помилок (Undo/Redo, Forcing Functions) та режиму «паніки» для екстремальних умов.
Очікувані результати навчання	• Вміння обґрунтовувати архітектурні рішення та структуру інтерфейсів (будь-якого типу – GUI, CLI, NUI, VUI) за допомогою математичних моделей та формальних критеріїв. • Проєктувати різнопланові інтерфейси взаємодії — від низькорівневих апаратних контрактів та мережевих API до складних графічних і природних оболонок на базі результатів наукових досліджень у галузі когнітивної та інженерної психології.

	• Застосовувати інструменти автоматизованої верифікації та тестування (OpenAPI, Pact, Playwright) для гарантування стабільності системних стиків. • Вміння керувати даними та станом системи – застосування стратегії Optimistic UI для покращення користувацького досвіду. • Вміння адаптувати рішення під різні платформи, враховуючи специфіку нативного рендерингу та обмежені ресурси IoT-пристроїв. • Вміння оцінювати інтерфейс за допомогою метрик юзабіліті та проводити повний цикл E2E-тестування.
Інформаційне забезпечення	Методичний посібник для лабораторних робіт Комплект презентацій
Види навчальних занять	Лекції (36 год), лабораторні заняття (18 год)
Вид семестрового контролю	диф. залік
Максимальна кількість здобувачів на семестр	Без обмежень

Декан факультету _____ *Ігор ГОМІЛКО*