



Проектування і розробка застосунків Smart-систем

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	Інформаційні технології
Спеціальність	F2 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці
Статус дисципліни	Вибіркова
Форма навчання	Очна (денна)
Рік підготовки, семестр	3-й курс, 2-й семестр
Обсяг дисципліни	4 кредити / 120 год. (лекцій 36 год., лаб. 18 год., СРС 66 год.)
Семестровий контроль/ контрольні заходи	Залік, Модульна контрольна робота
Розклад занять	http://schedule.kpi.ua/
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: Сарибога Ганна Володимирівна Практичні і: Сарибога Г.В.
Розміщення курсу	GitHub Classroom, eCampus

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Курс розкриває ключові технології проектування та розробки розумних пристроїв на базі мікроконтролерів. Студенти отримують знання про архітектуру мікроконтролерів, принципи їхнього програмування, інтеграцію сенсорів та акторів, протоколи передачі даних, а також способи інтеграції з хмарними платформами та мобільними застосунками. Особливий акцент робиться на практичному застосуванні цих технологій у створенні Smart-пристроїв для автоматизації, моніторингу та управління розумних систем.

Мета дисципліни

Надати студентам теоретичні знання та практичні навички для проектування та розробки застосунків Smart-систем із використанням сучасних технологій та інструментів, орієнтуючись на реальні прикладні задачі.

Предмет дисципліни

- Огляд архітектури мікроконтролерів та їх застосування в Smart-пристроях.
- Інтеграція сенсорів і акторів для створення автономних систем.

- Розробка програмного забезпечення для мікроконтролерів.
- Протоколи зв'язку та передачі даних для мікроконтролерів.
- Інтеграція розумних пристроїв із хмарними сервісами та мобільними додатками.

Результати навчання

Знання:

- Основи архітектури мікроконтролерів (ESP32, STM32, Arduino).
- Типи сенсорів і акторів, принципи їхньої роботи.
- Протоколи передачі даних (UART, I2C, SPI, MQTT).
- Методи інтеграції пристроїв із хмарними платформами (AWS IoT, Azure IoT).
- Особливості розробки розумних пристроїв для різних галузей (розумний дім, енергетика, транспорт).

Уміння:

- Програмувати мікроконтролери для роботи із сенсорами та акторами.
- Використовувати популярні фреймворки та середовища (Arduino IDE, PlatformIO).
- Реалізовувати зв'язок між пристроями через локальні мережі та Інтернет.
- Інтегрувати пристрої з мобільними додатками для моніторингу та управління.
- Використовувати протоколи для безпечної передачі даних між пристроями та сервісами.

2. Пререквізити та постреквізити дисципліни

Фахові компетентності:

ФК 14. Здатність проектувати та розробляти Smart-пристрої на базі мікроконтролерів.

ФК 17. Використання технологій Інтернету речей для інтеграції пристроїв у Smart-системи.

Програмні результати навчання:

ПРН 19. Розробляти програмне забезпечення для мікроконтролерів з урахуванням апаратних обмежень.

ПРН 24. Інтегрувати розумні пристрої з хмарними сервісами для збору та обробки даних.

Зміст навчальної дисципліни

Тема 1: Вступ до розумних пристроїв та мікроконтролерів

Тема 2: Робота із сенсорами та акторами

Тема 3: Протоколи зв'язку та обміну даними

Тема 4: Інтеграція з хмарними сервісами

Тема 5: Прикладні кейси та проекти Smart-пристроїв

4. Навчальні матеріали та ресурси

Базова література

1. Monk S., *Programming Arduino: Getting Started with Sketches*, McGraw-Hill, 2020.
2. Richardson M., Wallace S., *Getting Started with Raspberry Pi: Electronics Projects with Python, Scratch, and Linux*, O'Reilly Media, 2021.

3. Huggett C., *ESP32 Programming for Beginners: Programming Arduino and MicroPython for the ESP32 Board, Independently Published*, 2021.
4. Bannister A., *IoT Development for Beginners: From Sensors to Systems*, Apress, 2020.

Додаткова література

1. Margolis M., *Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects*, O'Reilly Media, 2020.
2. Petrus D., *Building Smart IoT Projects: Using Big Data and Machine Learning on Internet of Things Applications*, Packt Publishing, 2021.
3. Althoff J., *Exploring Raspberry Pi: Interfacing to the Real World with Embedded Linux*, Wiley, 2020.
4. Офіційна документація Arduino – <https://www.arduino.cc/reference/en/>
5. Документація MQTT – <https://mqtt.org/documentation>

• **Навчальний контент**

5. Методика опанування навчальної дисципліни (освітнього компонента)

Тематика лекційних занять.

Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на інформаційні джерела)
Тема 1: Вступ до розумних пристроїв та мікроконтролерів Лекція 1: Огляд Smart-пристроїв: концепції, сфери застосування. Лекція 2: Архітектура мікроконтролерів: основні компоненти (ESP32, STM32, Arduino). Лекція 3: Мікроконтролери та їх програмування: інструменти та середовища розробки. Лекція 4: Мережеві архітектури для розумних пристроїв: локальні мережі та Інтернет речей. Тема 2: Робота із сенсорами та акторами Лекція 5: Типи сенсорів та їх характеристики (температура, вологість, рух). Лекція 6: Актори: реле, мотори, LED-дисплеї. Лекція 7: Принципи роботи з сенсорами: читання даних, калібрування. Лекція 8: Управління акторами: основні алгоритми. Тема 3: Протоколи зв'язку та обміну даними

Лекція 9: Провідні протоколи зв'язку (UART, I2C, SPI): порівняння, застосування.

Лекція 10: Бездротові протоколи (Wi-Fi, Bluetooth, Zigbee): огляд, переваги.

Лекція 11: MQTT як протокол обміну даними для IoT.

Лекція 12: Забезпечення безпеки передачі даних.

Тема 4: Інтеграція з хмарними сервісами (4 лекції, 2 практичні заняття)

Лекція 13: Хмарні платформи для IoT (AWS IoT, Azure IoT, Google Cloud IoT).

Лекція 14: Збір даних у хмарі: принципи та застосування.

Лекція 15: Віддалене управління пристроями через хмарні сервіси.

Лекція 16: Моніторинг і обробка даних у реальному часі.

Тема 5: Прикладні кейси та проєкти Smart-пристроїв

Лекція 17: Розумний дім: автоматизація, безпека, енергозбереження.

Лекція 18: Інтелектуальні системи у транспорті та енергетиці.

Тематика практичних занять.

№	Назва практичної роботи	Основні завдання / зміст
1	Ознайомлення з архітектурою Smart-пристроїв	Аналіз основних типів Smart-пристроїв. Створення блок-схеми системи "сенсор–контролер–актор–мережа". Ознайомлення з платами ESP32, Arduino, STM32 у симуляторах Tinkercad / Wokwi / Proteus.
2	Програмування мікроконтролера Arduino/ESP32	Написання першої програми (миготіння LED, delay, loop). Завантаження та налагодження через Arduino IDE / PlatformIO. Демонстрація роботи програми у симуляторі.

3	Зчитування даних із сенсорів	Підключення цифрових/аналогових сенсорів (температура, вологість, освітленість). Написання коду читання значень і відображення у серійному моніторі. Використання датчиків DHT22, BH1750, LM35.
4	Керування акторами	Підключення реле, сервоприводів, світлодіодних індикаторів. Реалізація простих алгоритмів вмикання/вимикання на основі даних сенсорів.
5	Передача даних між пристроями через UART/I2C/SPI	Порівняння інтерфейсів. Реалізація передачі даних між двома мікроконтролерами або сенсором і дисплеєм. Вивчення команд, швидкості обміну, синхронізації.
6	Передача даних по Wi-Fi (ESP32)	Підключення ESP32 до локальної мережі Wi-Fi. Надсилання даних сенсорів через HTTP-запит до веб-інтерфейсу (наприклад, Thingspeak або власний сервер Flask/Streamlit).
7	Публікація даних через MQTT-брокер	Підключення ESP32 до MQTT-брокера (Mosquitto / HiveMQ). Створення топіків для передачі та прийому даних. Візуалізація у Node-RED.
8	Інтеграція з хмарним сервісом	Реєстрація пристрою у хмарній платформі (ThingsBoard / Firebase / Google Cloud IoT). Надсилання та зберігання даних у хмарі. Побудова графіків у реальному часі.
9	Міні-проект Smart-пристрою	Розробка та демонстрація повного Smart-рішення: сенсори → обробка → актори → передача даних → хмара → веб-інтерфейс. Приклади: “Розумна теплиця”, “Енергоефективне освітлення”, “Моніторинг мікроклімату”.

6. Самостійна робота студента

На самостійну роботу студента відведено 66 годин.

Перелік питань для самостійного опрацювання

1. Поняття Smart-систем та їх архітектура.
2. Основні апаратні та програмні компоненти Smart-систем.

3. Використання мікроконтролерів у Smart-системах: огляд популярних моделей (ESP32, Arduino, STM32).
4. Хмарні, Edge та Fog обчислення в Smart-застосунках.
5. Протоколи зв'язку: MQTT, CoAP, HTTP.
6. Інтеграція IoT та штучного інтелекту.
7. Забезпечення безпеки та приватності даних.
8. Платформи для розробки Smart-застосунків: огляд і вибір.
9. Програмування мікроконтролерів для Smart-систем (основи прошивки та налаштування).
10. Інтерфейси зв'язку мікроконтролерів (UART, SPI, I2C, GPIO).

7. Контрольна робота

Метою контрольної роботи є закріплення та перевірка теоретичних знань із освітнього компонента, набуття студентами практичних навичок самостійного вирішення. Модульна контрольна робота (МКР) виконується після вивчення Тем 5. Контрольна робота проводиться у середовищі G Suite Classroom. Кожен студент отримує індивідуальне завдання.

• Політика і контроль

8. Політика навчальної дисципліни (освітнього компонента)

Система вимог під час вивчення освітнього компонента:

- правила відвідування занять: заборонено оцінювати присутність або відсутність здобувача на аудиторному занятті, в тому числі нараховувати заохочувальні або штрафні бали. Відповідно до РСО даної дисципліни бали нараховують за відповідні види навчальної активності на лекційних та практичних заняттях.
- політика термінів виконання робіт та перескладань: якщо студент не проходив або не з'явився на МКР (без поважної причини), його результат оцінюється у 0 балів. Перескладання результатів МКР не передбачено;
- політика щодо академічної доброчесності: Кодекс честі Національного технічного університету України «Київський політехнічний інститут» <https://kpi.ua/files/honorcode.pdf> встановлює загальні моральні принципи, правила етичної поведінки осіб та передбачає політику академічної доброчесності для осіб, що працюють і навчаються в університеті, якими вони мають керуватись у своїй діяльності, в тому числі при вивченні та складанні контрольних заходів з дисципліни «Технології збору та обробки даних»;
- при використанні цифрових засобів зв'язку з викладачем (мобільний зв'язок, електронна пошта, переписка на форумах та у соцмережах тощо) необхідно дотримуватись загальноприйнятих етичних норм, зокрема бути ввічливим та обмежувати спілкування робочим часом викладача.

• 9. Види контролю та рейтингова система оцінювання результатів навчання (РСО)

Поточний контроль: тестування, МКР, виконання та захист практичних робіт.

Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог силлабусу.

Семестровий контроль: залік

Умови допуску до семестрового контролю: семестровий рейтинг 40 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно

94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

Загальна рейтингова оцінка студента після завершення семестру складається з балів, отриманих за:

- виконання завдань до практичних занять (90 балів);
- виконання модульної контрольної роботи (МКР) (10 балів);

Тестування за темами лекцій

На лекціях може бути проведений експрес-контроль знань студентів. Тестування проводяться на довільних лекціях 2 рази протягом семестру, під час лекції. Практичні заняття Оцінюються 9 практичних робіт. Максимальний ваговий бал rПР =90.

Терміни виконання лабораторних робіт:

Практична робота № 1, 2, 3 – до першого календарного контролю

Практична робота № 4,5 – до другого календарного контролю

Практична робота № 6 – до залікового тижня

Оцінювання практичних робіт:

- робота, виконана невчасно, не зараховується;
- робота, виконана не самостійно, не зараховується

УВАГА! Умовою допуску до заліку є рейтинг не менше 45 балів.

Студенти, які на момент заліку не мають 45 балів за поточним контролем, не допускаються до основної складання заліку та готуються до перескладання. УВАГА! Для допуску до перескладання заліку необхідно у визначений викладачем термін здати заборгованості по практичних роботах.

Модульний контроль

На одному з лекційних занять проводиться модульна контрольна робота після вивчення Темі 2. Контрольна робота проводиться у середовищі G Suite Classroom. Кожен студент отримує індивідуальне завдання. Максимальний ваговий бал rМКР = 10

Календарний контроль

Календарний контроль базується на поточній рейтинговій оцінці. Умовою позитивної атестації є значення поточного рейтингу студента не менше 50% від максимально можливого на час атестації. Бал, необхідний для отримання позитивного календарного контролю доводиться до відома студентів викладачем не пізніше ніж за 2 тижні до початку календарного контролю.

Форма семестрового контролю – залік

Максимальна сума балів складає 100. Необхідною умовою допуску до заліку є рейтинг не менше 40 балів. Для отримання заліку з кредитного модулю «автоматом» потрібно мати рейтинг не менше 60 балів. Студенти, які наприкінці семестру мають рейтинг менше 60 балів, а також ті, хто хоче підвищити свою оцінку в системі ECTS, виконують

залікову контрольну роботу. При цьому набрані бали студентом анулюються, а оцінка за залікову контрольну роботу є остаточною. Залікова робота. Залікова робота проводиться на останньому лекційному занятті. Студент проходить тестування у середовищі Moodle. На тестування пропонується 100 тестових питань, кожне з яких оцінюється в 1 бал. Для отримання позитивної оцінки необхідно набрати 60 балів і вище. Час тестування зазвичай складає 100 хвилин, але може бути скоригований лектором та (або) викладачем, який приймає залік.

10. Додаткова інформація з дисципліни (освітнього компонента)

Перелік питань, які виносяться на семестровий контроль:

Перелік питань для семестрового контролю

Розширений перелік питань для семестрового контролю

1. Основні поняття IoT та приклади Smart-систем.
2. Архітектура IoT-систем: рівні та компоненти.
3. Огляд апаратних платформ для IoT (ESP32, Arduino).
4. Програмні середовища для розробки IoT-застосунків.
5. Протоколи передачі даних у IoT: особливості MQTT, HTTP, WebSocket.
6. Способи підключення сенсорів до мікроконтролерів (UART, I2C, GPIO).
7. Налаштування з'єднання IoT-пристроїв з Wi-Fi мережею.
8. Реалізація HTTP-запитів: GET і POST методи.
9. Інтеграція IoT-систем із хмарними сервісами (Firebase, ThingsBoard).
10. Візуалізація даних у реальному часі: Node-RED, Chart.js.
11. Публікація та підписка в MQTT-протоколі.
12. Віддалене керування пристроями через веб-сервіси.
13. Віртуальне моделювання IoT-проектів у Tinkercad і Proteus.
14. Методи забезпечення безпеки IoT: шифрування, автентифікація.
15. Оптимізація енергоспоживання мікроконтролерів (режими сну).
16. Аналіз помилок і стабільності IoT-систем.
17. Розробка сценаріїв автоматизації в Smart-системах.
18. Використання Node-RED для створення панелей керування.
19. Побудова цифрових двійників IoT-систем.
20. Практичне застосування IoT у реальних умовах: Smart Home, Smart City, Smart Grid.

Робочу програму навчальної дисципліни (силабус):

Складено ст. викл. Сарибогою Ганною Володимирівною

Ухвалено кафедрою ІПЗЕ (протокол № 34 від 10.05.2025)

Погоджено Методичною комісією факультету (протокол № 9 від 31.05.2025)