Embedded System (CEDT) (Draft)

ให้ตัวแทนกลุ่มแจ้งข้อมูล พวกสมาชิกและหัวข้อได้เลยนะครับ
https://www.mycourseville.com/?q=courseville/worksheet/57052/1455443

ข้อมูลกลุ่มและคะแนนโครงงานล่าสุด
https://docs.google.com/spreadsheets/d/1KBOXe_JOe-j6VeMh6DcLyExU8ztRGeX-TQrY4JHJJYo

Final Project 2024: Smart Devices

1 Objectives

In the final project, you get a chance to exercise your knowledge, creativity and management skill to create a contactless device. In particular, you will be using ESP32 to read/write data to/from actuators/sensors. From there, you can connect to the Internet and create a smart device! For the Internet communication, you can use any API you would like to use. We recommend NetPIE. However, you may find some other API like Firebase easier to get start.

You can choose what you want your smart device to do under the theme smart devices. There are various examples of IoT devices such as monitoring temperature, avoiding heat stroke, reducing waste, and so on. You can consult us in case you need any kind of additional sensors for your project.

You will also be developing team works skill by working in a group of **four**. Each one of you must assign yourself to one of the following role: system architecture, front-end development, embedded system development, and UI designer and development.

2 Background

2.1 Smart Devices

A smart device is an electronic device, generally connected to other devices or networks via different wireless protocols (such as Bluetooth, Zigbee, near-field communication, Wi-Fi, NearLink, Li-Fi, or 5G) that can operate to some extent interactively and autonomously. Several notable types of smart devices are smartphones, smart speakers, smart cars, smart thermostats, smart doorbells, smart locks, smart refrigerators, phablets and tablets, smartwatches, smart bands, smart keychains, smart glasses, and many others. The term can also refer to a device that exhibits some properties of ubiquitous computing, including—although not necessarily—machine learning. [Excerpt from Wikipedia]

2.2 NETPIE

As state above, you may use other framework.

NECTEC has developed the cloud-based IoT platform-as-a-service named NETPIE (https://netpie.io) and launched the service since September 2015. NETPIE is a cloud platform that interconnects IoT devices together in the most seamless and transparent manner. By moving the complexity of connecting IoT devices from application developers or device manufacturers to the cloud, NETPIE helps shorten development time and reduce burden of server administration.

(This is an assert from https://www.nectec.or.th/en/innovation/service-innovation/netpie.html. Please see the link for more information)

From your point of view, you can connect to NETPIE through NodeMCU. NETPIE already provides a simple interface for connecting NodeMCU to NETPIE. Please see https://netpie.io/ for more details.

Note that there are two versions of NetPIE API. If you look for tutorials online, it may only be applicable to older version of the API.

3 Groups

We expect you to work in groups and each person has a different roles. Here is an example of how the groups may organized.

1. System Architecture - Design and integrated the system
2. Embedded System Development - Develop code on the MCU
3. UI/UX Designer and Development - Design and develop user interface
4. Team Management - Manage team to work in the time scale.

Note that you may use a different structure. However, we encourage the group that each person has a different role.

4 Requirement and Submission

The tentative date is after the final exam week. Each group must prepare 10 minutes demonstration.

Your application must contains at least the following features:

1. Using communication to sensors/actuators
2. Connected to Internet and Cloud System
3. Justification to create your smart devices.

Each week, you must submit a short progress report (a paragraph) on MCV.

On the day of submission, you will be prepare the following:

1. 10 minutes presentation describing how your devices behave.

2. At least 4 pages report final project. Each team member must describes at least in one page about their role and responsibility.
3. Source Code: During the course of development, you must use a repository, such as Git, Mercury or SubVersion. The repository must be accessible from the Internet. You must submit the archive of the source code along with the link to repository.

5 Marking guide

I. Problem Definition (5 points)

- Relevance and Impact (3 points): Clearly describes a real-world problem affecting a significant group of people and justifies its urgency. The problem should be well-defined and appropriately scoped for an embedded systems project.

- Adequate Detail (2 points): Provides sufficient background information, context, and relevant data to understand the problem and the proposed solution.

II. System Design and Functionality (10 points)

- Sensor and Actuator Integration (5 points):

  Utilizes at least 5 different sensor types. (Diversity is encouraged.)

  Incorporates at least 1 sensor utilizing a microphone or camera.

  Integrates at least 2 sensors from Sensor Node platforms.

  Integrates at least 2 sensors from the Gateway platform.

  Integrates at least 1 actuator (motor, pump, etc).

- Data Flow and Processing (5 points):

  Implements wire/wireless data transmission from Arduino/Sensor Node to ESP32.

  Performs real-time data processing on ESP32 for immediate control actions.

  Receives data from a camera or microphone via computer/mobile device and utilizes AI packages for decision-making and output.

  Demonstrates scalability by simulating support for multiple sensor nodes.

  Uploads raw data to at least two different cloud storage platforms (e.g., Blynk, Google Sheets, ThingSpeak, Firebase).

III. Implementation and Testing (30 points)

- Functionality (10 points):

Basic Functionality (5 points): Core features of the system are partially implemented and demonstrable.

Complete Functionality (+5 points): All designed features are fully implemented and working as intended.

- Quality of Implementation (20 points):

  Field Testing (3 points): System undergoes real-world testing with continuous data acquisition and actual users. Bonus points for deployment in multiple locations.

  Robustness (3 points): System demonstrates reasonable durability and stability.

  Organization and Aesthetics (3 points): Wiring, sensors, and components are neatly organized. The project exhibits a polished and user-friendly design.

  Portability (3 points): System can be easily transported and set up without damage.

  User Interface (2 points): Displays information in a clear, intuitive, and aesthetically pleasing manner on chosen platforms (mobile app, web app).

  Usability (2 points): System is intuitive and easy to use, requiring minimal instruction.

  Safety (2 points): System operates safely and minimizes potential risks to users and the environment.

  Alert System (Bonus 2 points): Implements a notification system (e.g., LINE, email) to alert users of important events or system status.

IV. Testing and Analysis (5 points)

- Testing (2 points): Thorough testing procedures are documented, demonstrating system functionality and performance evaluation under various conditions.

- Analysis and Comparison (3 points): Results are analyzed and compared against project objectives and existing similar systems. Performance metrics are used to evaluate the effectiveness of the solution.

V. Presentation and Documentation (20 points)

- Presentation (10 points): (presenter may be randomly selected by examiner, every member should prepare)

Clarity and Organization (3 points): Presentation follows a logical flow and effectively communicates the project's purpose, design, and results.

Delivery (3 points): Engaging presentation style with clear voice projection, appropriate body language, and minimal reliance on reading slides.

Slides (2 points): Slides are visually appealing, well-organized, and effectively support the presentation content.

Demonstration (2 points): Live demonstration of the project with a clear plan and explanation of the setup and procedures.

- Video Documentation (5 points):

  Content (3 points): Video clearly identifies team members, university affiliation, and provides a concise project overview. It demonstrates the system's functionality and explains key components.

  Quality (2 points): Video is well-produced, visually appealing, and effectively communicates the project's essence. It is uploaded to YouTube, Facebook, TikTok, etc.

- Documentation (5 points):
  Note more than 10 pages
  Introduction and problem statement
  Related work or similar solution
  System Design
    - Overview diagram
    - Sensor node
    - Gateway
    - Cloud / Storage
    - Dashboard

  Test result
  Discussion and conclusion

VI Team

- Groups with more than 4 members will incur a penalty of -2.5 points per additional member.

Total: 70 points

Timeline

11 November	Project release
2 December	Presentation

Information from 4 invited companies

- SynTech
- Sharp
- CPF
- SC Asset

https://drive.google.com/drive/folders/1N5UgUTyMlFtlNxQgJTJFNBwh017ux6bi?usp=sharing

FAQ นิสิตสามารถสอบถามคำถามเพิ่มเติมได้เลย

1 อาจารย์จะมี sensor เพิ่มเติมให้ไหมคะ หรือต้องซื้ออุปกรณ์กันเองอะคะ

ซื้อเพิ่มเติมเองครับ

2 sensor จาก Sensor Node platforms กับ ESP32 platform. ต่างกันยังไงครับ
อยากให้มีการส่งข้อมูลระหว่าง บอร์ด 2 ตัวครับ
ตัวนึงเลยทำหน้าที่เป็น sensor node อีกตัวทำหน้าที่เป็น gateway
ซึ่งถ้าเราใช้ ESP32 1 ตัว เป็น sensor node และอีกตัวเป็น gateway ก็ได้
หรือจะใช้ บอร์ดแบบอื่นก็ได้

3 document ควรมีอะไรบ้างครับ พอมี template มั้ยครับ
Introduction
Related work
Methodology
Result
Conclusion

4 ตรงหัวข้อที่ต้องใช้ AI ประมวลผล (กล้องหรือไมโครโฟน) สามารถใช้ Open-source ได้เลยไหมครับ หรือจำเป็นต้องเทรน Model เอง
Open source ได้เลยครับ