



UNIVERSIDAD
COMPLUTENSE
MADRID

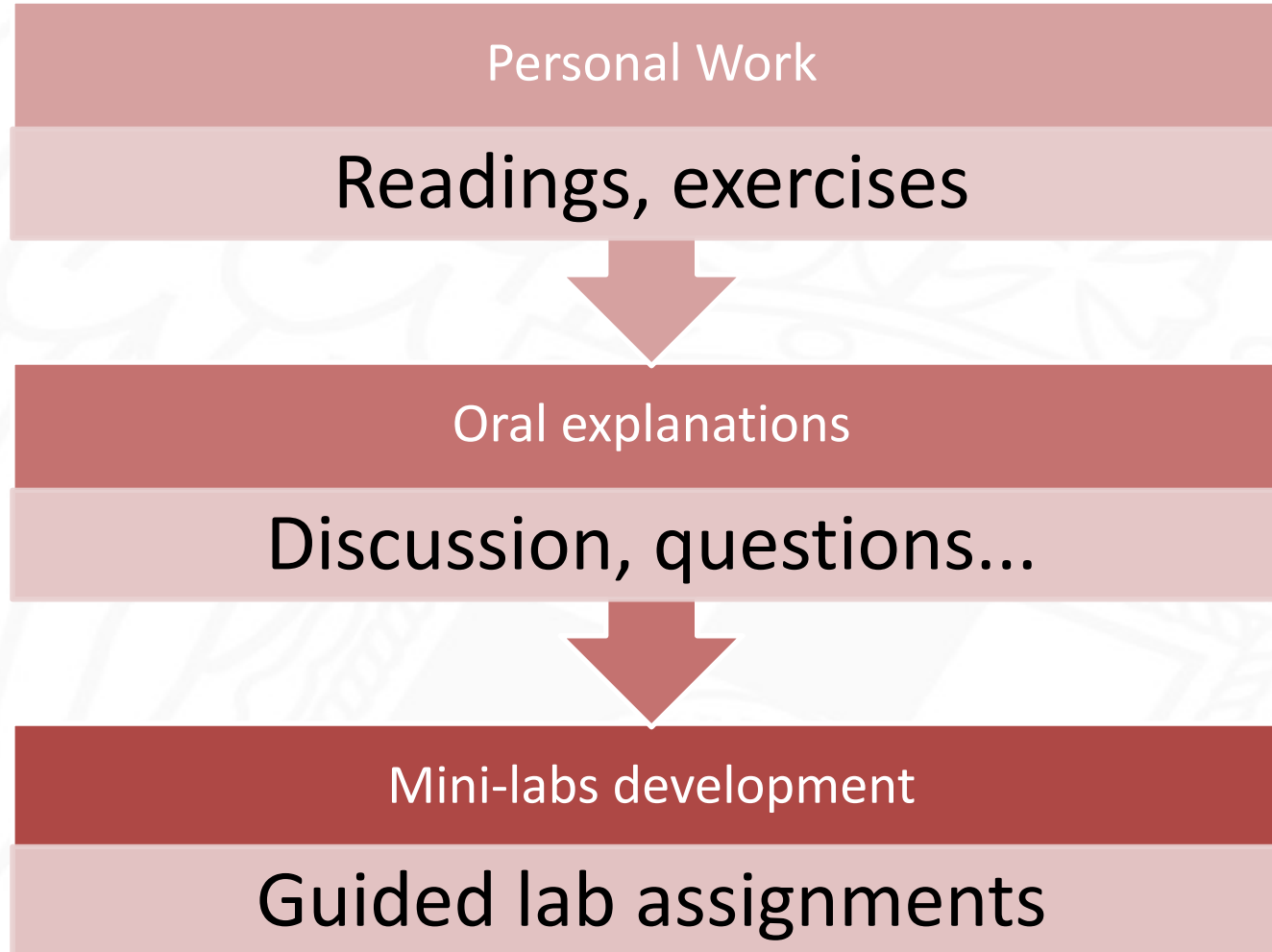
IoT Node Architecture

José Ignacio Gómez
Katzalín Olcoz

- ❑ José Ignacio Gómez Pérez
 - jigomez@ucm.es
- ❑ Katzalín Olcoz
 - katzalin@ucm.es
- ❑ Contact us with any question

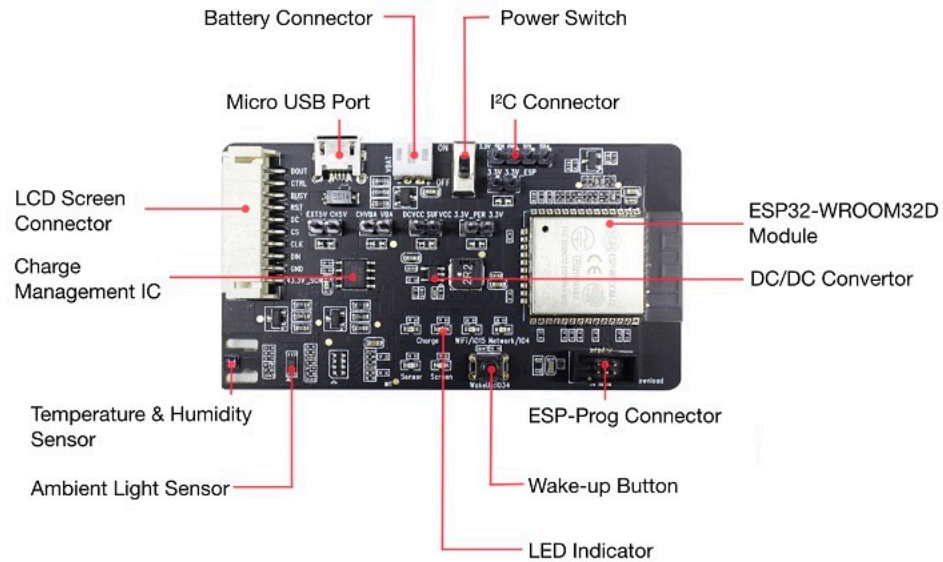
1. Get a global view of IoT applications lifecycle
2. Get to know the main components of an IoT device
3. Learn to use cross-development and debugging tools
4. Understand the specificities of Real Time Operating Systems (RTOS)
5. Use the most relevant serial communication interfaces
6. Use a variety of sensors
7. Understand the problems of data acquisition
8. Learn the main problems concerning energy consumption and supply
9. Firmware update basics

□ Practice oriented

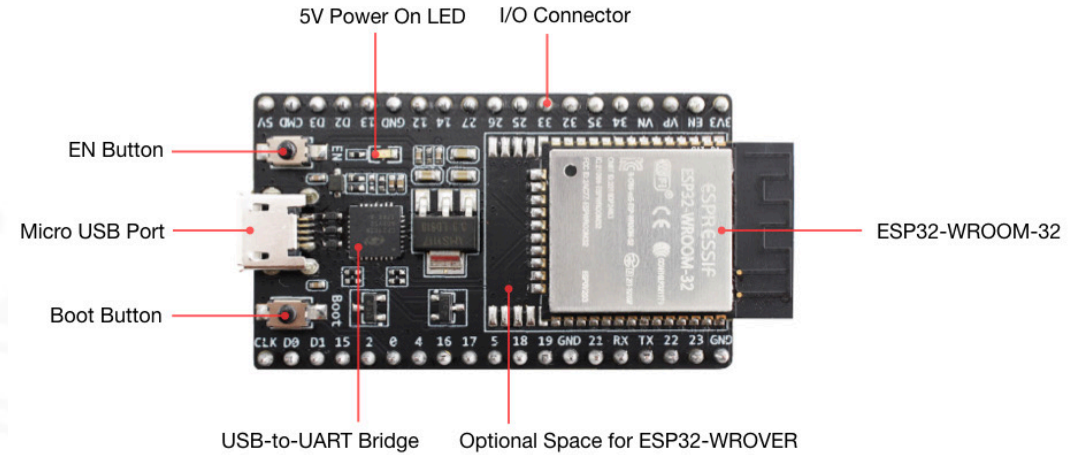


CONTENTS

1. IoT ecosystem
2. IoT node lifecycle
3. Microcontrollers, *System-on-chips* (SoC), development boards



ESP32 – Meshkit Sense



ESP32 - DevKitC

CONTENTS

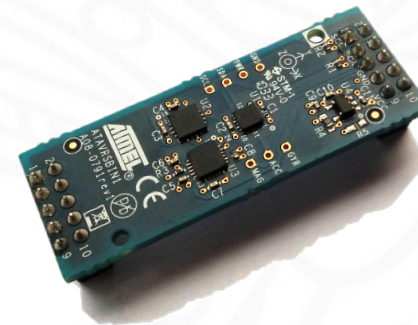
1. Cross-compilation and debugging frameworks
2. Operating Systems
 - FreeRTOS (ESP-IDF)
3. File systems. Logging



By HikingArtist.com

CONTENTS

1. Sensors
2. Simple devices: timers, GPIO, interruptions
3. Serial bus connections: I2C, SPI, UART
4. ADC / DAC basics
5. Periodic sampling and application structure
6. Energy consumption and device powering



- ❑ McEwen, A. and Cassimally, H. (2014) *Designing the Internet of Things*, John Wiley & Sons, Ltd, Chichester, 1.
- ❑ Qusay F. Hassan (2018) *Internet of Things A to Z. Technologies and Applications*, IEEE Press
 - Specially Chapter 6
- ❑ S. Cirani et al. (2019) *Internet of Things. Architectures, Protocols and Standards* . Wiley & Sons Ltd.

❑ Lab assignments.

- Every student must finish the programming projects before the specified deadline
- This will account for 40% of overall grade

❑ Personal *paper project*

- IoT project proposal
- Paper project, no development involved
- 30% of overall grade

❑ Personal development project.

- Every student must develop a personal project
- You must use the provided hardware, but can include new devices if you want
- The project must
 - Periodically read several I2C sensors
 - Log into flash relevant events
 - Only allow signed binaries to run
- 30% of overall grade

- ❑ Please ask when you need it. I would like it to be an **interactive** class