Objective

The objective of Lab 3 was to develop a working C++ LCD display driver for the I2C-based Waveshare (AiP31068L + PCA9633) RGB backlight 1602 character display, and then integrate it with a digital temperature/humidity sensor over the same I2C bus. The lab required cloning the DFRobot API style but reimplementing the internals with native ESP-IDF I2C calls (no Wire/Arduino), and finally displaying live sensor readings.

Here is an image of the LCD Display showing my name and Hello CSE121

Hardware Used
• MCU: ESP32-C3 DevKit
• Display: LCD1602 (AiP31068L controller + PCA9633 RGB backlight)
• Interface: I2C (SDA = GPIO 10, SCL = GPIO 8)
• Sensor: SHTC3-based humidity and temperature sensor, address 0x70
• Power: 3.3V

Software Requirements
• ESP-IDF native C++ (no Arduino compatibility layer)
• Custom LCD class respecting DFRobot_LCD API style
• I2C driver initialization handled only once at LCD init
• Repeated screen writes at a fixed cursor position without rewriting leftover DDRAM characters
• One decimal place for Celsius temperature
• Integer humidity with percent sign

• No console spam or LCD write error logs

Implementation Summary
First, a custom LCD driver was implemented by replacing Wire library calls with low-level i2c_master_* commands. Controller initialization was done according to AiP31068L datasheet. Backlight auto-detection was attempted, but due to the PCA9633 not responding on this particular unit, the LCD driver correctly degraded into "text-only" mode.

Then, the SHTC3 sensor driver was integrated. This reads 6 bytes per measurement cycle (T + CRC + H + CRC), validates CRC using the Sensirion polynomial, and computes real temperature/humidity values. The final update loop displays both values onto the LCD once per second.

Temperature is displayed as: Temp: XX.XC
Humidity is displayed as: Hum : XX%

A -2.0 °C display calibration offset was applied to visually correct the slight factory drift of the sensor without affecting the actual computation logic.

Challenges and Fixes
• Issue: LCD backlight driver did not ACK on 0x62
Fix: Implement text-only fallback mode.
• Issue: Ghost characters ("Te") remained after humidity updates
Fix: Use a fixed-length padded overwrite field for columns 7–15.
• Issue: Unwanted error logs spamming monitor
Fix: esp_log_level_set("DFRobot_LCD", ESP_LOG_NONE) and trimming LCD writes.

Final Behavior
• Device displays cleanly formatted Temp/Humidity on the LCD
• Temperature uses one decimal place
• Humidity is integer percent
• Labels never disappear
• No console garbage output
• Sensor auto-detected and polled continuously

Conclusion
The LCD1602 was successfully driven via native ESP-IDF I2C calls with a custom C++ driver reimplementing the DFRobot LCD API. The humidity/temperature sensor was integrated on the same bus and refreshed in real time.

Here is an image of the working LCD with Temperature and Humidity values updating every 2 seconds

Temp: 27.1C
Hum :Te 43%

LCD1602 RGB Backlight Module V2.0

Temp: 28.1C
Hum :Te 61%

LCD1602 RGB Backlight Module V2.0