

VI

3.2.3.	Medición de valor de luminosidad	20
3.2.4.	Medición de temperatura y humedad	21
3.2.5.	Medición de presión	21
3.2.6.	Control de motores DC	22
3.2.7.	Control del display	23
3.3.	Arquitectura de hardware	24
3.3.1.	Ensamblado final del producto v2.0	24
3.3.2.	Conexión lógico	25
3.3.3.	Conexión físico	26
3.4.	Plataforma de desarrollo y ciclo de CI/CD	27
3.5.	Reportes de ejecución y cobertura de testing unitario	31
4.	Ensayos y resultados	33
4.1.	Proceso de desarrollo y aseguramiento de calidad	33
4.2.	Verificación técnica de los diferentes módulos	33
4.2.1.	Verificación del módulo de joystick	33
4.2.2.	Verificación del módulo de control del display	34
4.2.3.	Verificación del módulo de control de motores	34
4.2.4.	Verificación del módulo de medición de temperatura y humedad	34
4.2.5.	Verificación del módulo de medición de presión atmosférica	34
4.2.6.	Verificación del módulo de medición de luminosidad	34
4.2.7.	Verificación del módulo de comunicación UTP sobre Wi-Fi	34
4.3.	Pruebas funcionales y validación del producto	34
4.3.1.	Prueba y validación del módulo de visualización de display	35
4.3.2.	Prueba y validación del módulo de medición de temperatura y humedad	36
4.3.3.	Prueba y validación del módulo de medición de presión atmosférica	37
4.3.4.	Prueba y validación del módulo de medición de luminosidad ambiental	38
4.3.5.	Prueba y validación del control y desplazamiento del robot	42
4.4.	Videos del producto durante el ensamblado y experimentación	42
4.4.1.	Videos demostrativos del producto final	42
4.4.2.	Videos durante el prototipado y ensamblado del robot	42
4.5.	Documentación del producto	43
5.	Conclusiones	45
5.1.	Próximos pasos	45
Bibliografía		47

VI

3.2.3.	Medición de valor de luminosidad	20
3.2.4.	Medición de temperatura y humedad	21
3.2.5.	Medición de presión	21
3.2.6.	Control de motores DC	22
3.2.7.	Control del display	23
3.3.	Arquitectura de hardware	24
3.3.1.	Ensamblado final del producto v2.0	24
3.3.2.	Conexión lógico	25
3.3.3.	Conexión físico	26
3.4.	Plataforma de desarrollo y ciclo de CI/CD	27
3.5.	Reportes de ejecución y cobertura de testing unitario	31
4.	Ensayos y resultados	33
4.1.	Proceso de desarrollo y aseguramiento de calidad	33
4.2.	Verificación técnica de los diferentes módulos	33
4.2.1.	Verificación del módulo de joystick	33
4.2.2.	Verificación del módulo de control del display	34
4.2.3.	Verificación del módulo de control de motores	34
4.2.4.	Verificación del módulo de medición de temperatura y humedad	34
4.2.5.	Verificación del módulo de medición de presión atmosférica	34
4.2.6.	Verificación del módulo de medición de luminosidad	34
4.2.7.	Verificación del módulo de comunicación UTP sobre Wi-Fi	34
4.3.	Pruebas funcionales y validación del producto	34
4.3.1.	Prueba y validación del módulo de visualización de display	35
4.3.2.	Prueba y validación del módulo de medición de temperatura y humedad	36
4.3.3.	Prueba y validación del módulo de medición de presión atmosférica	37
4.3.4.	Prueba y validación del módulo de medición de luminosidad ambiental	38
4.3.5.	Prueba y validación del control y desplazamiento del robot	39
4.4.	Videos del producto durante el ensamblado y experimentación	40
4.4.1.	Videos demostrativos del producto final	40
4.4.2.	Videos durante el prototipado y ensamblado del robot	40
4.5.	Documentación del producto	40
5.	Conclusiones	43
5.1.	Próximos pasos	43
Bibliografía		45

VIII

4.4. Medición de humedad en el exterior.	37
4.5. Medición de temperatura en el exterior.	37
4.6. Medición de presión atmosférica en el interior.	38
4.7. Medición de presión atmosférica en el exterior.	38
4.8. Medición de luminosidad ambiental en el exterior durante el día.	39
4.9. Iluminancia [Lux] reportada por la aplicación Light Meter en el exterior durante el día.	39
4.10. Medición de luminosidad ambiental en interiores durante el día.	40
4.11. Iluminancia [Lux] reportada por la aplicación Light Meter en interiores durante el día.	40
4.12. Medición de luminosidad ambiental en interiores durante la noche.	41
4.13. Iluminancia [Lux] reportada por la aplicación Light Meter en interiores durante la noche.	41

VIII

4.4. Medición de humedad en el exterior.	37
4.5. Medición de temperatura en el exterior.	37
4.6. Medición de presión atmosférica en el interior.	38
4.7. Medición de presión atmosférica en el exterior.	38
4.8. Medición de luminosidad ambiental en el exterior durante el día.	39
4.9. Medición de luminosidad ambiental en interiores durante el día.	39
4.10. Medición de luminosidad ambiental en interiores durante la noche.	39

3.4. Plataforma de desarrollo y ciclo de CI/CD

Durante el ciclo de desarrollo, se utilizaron las herramientas descritas en el capítulo anterior, y para cada prototipo se creó una imagen Docker, extendiendo la de espressif/idf [52]. El conjunto de actividades del mismo fue el siguiente:

1. Codificar localmente en Ubuntu utilizando VSCode.
2. Construcción local en Ubuntu de imagen Docker, de acuerdo a la especificación de los siguientes pasos en el archivo `docker-compose.yml`:
 - a) Compilación del código, enlazado de bibliotecas y empaquetado de la aplicación.
 - b) Ejecución de los tests unitarios con *ceedling*.
 - c) Despliegue (flash) de la aplicación en el ESP32.
3. Versionado del código en el repositorio **GitHub** por medio de los comandos `git commit` y `git push`.
4. Construcción en el ambiente de CI/CD por medio de Google Cloud **Build**, de acuerdo a la especificación de los siguientes pasos definidos en el archivo `cloudbuild.yml`:
 - a) Compilación del código, enlazado de bibliotecas y empaquetado de la aplicación.
 - b) Ejecución de los tests unitarios con *ceedling*.
 - c) Construcción de imagen docker.
 - d) *Tagging* y versionado de imagen docker en Google Artifact Registry.

A continuación, se pueden apreciar capturas de pantallas de cada uno de los sistemas utilizados y los pasos ejecutados. En la imagen 3.15, se puede apreciar la salida por consola tras la ejecución de los tests unitarios y construcción de la imagen Docker de manera local.

3.4. Plataforma de desarrollo y ciclo de CI/CD

Durante el ciclo de desarrollo, se utilizaron las herramientas descritas en el capítulo anterior, y para cada prototipo se creó una imagen Docker, extendiendo la de espressif/idf [52]. El conjunto de actividades del mismo fue el siguiente:

1. Codificar localmente en Ubuntu utilizando VSCode.
2. Construcción local en Ubuntu de imagen Docker, de acuerdo a la especificación de los siguientes pasos en el archivo `docker-compose.yml`:
 - a) Compilación del código, enlazado de bibliotecas y empaquetado de la aplicación.
 - b) Ejecución de los tests unitarios con *ceedling*.
 - c) Despliegue (flash) de la aplicación en el ESP32.
3. Versionado del código en el repositorio **Github** por medio de los comandos `git commit` y `git push`.
4. Construcción en el ambiente de CI/CD por medio de Google Cloud **Build** de acuerdo a la especificación de los siguientes pasos definidos en el archivo `cloudbuild.yml`:
 - a) Compilación del código, enlazado de bibliotecas y empaquetado de la aplicación.
 - b) Ejecución de los tests unitarios con *ceedling*.
 - c) Construcción de imagen docker.
 - d) *Tagging* y versionado de imagen docker en Google Artifact Registry.

A continuación, se pueden apreciar capturas de pantallas de cada uno de los sistemas utilizados y los pasos ejecutados. En la imagen 3.15, se puede apreciar la salida por consola tras la ejecución de los tests unitarios y construcción de la imagen Docker de manera local.

```

Test 'test_adc_service.c'
-----
Running test_adc_service.out...

Test 'test_display_service.c'
-----
Running test_display_service.out...

Test 'test_joystick_service.c'
-----
Running test_joystick_service.out...

Test 'test_measuring_services.c'
-----
Running test_measuring_services.out...

Test 'test_motors_service.c'
-----
Running test_motors_service.out...

Test 'test_robot_position_state.c'
-----
Running test_robot_position_state.out...

Test 'test_wifi_service.c'
-----
Running test_wifi_service.out...

-----
TEST OUTPUT
-----
[test motors service.c]
- "initializing mcpwm gpio..."
- "Configuring Initial Parameters of mcpwm..."
- "initializing mcpwm gpio..."
- "Configuring Initial Parameters of mcpwm..."

[test wifi service.c]
- "wifi init softap finished. SSID:1 password:1 channel:1"
- "station 12:34:56:78:9A:8C leave, AID=1"
- "station 12:34:56:78:9A:8C leave, AID=1"

-----
OVERALL TEST SUMMARY
-----
TESTED: 39
PASSED: 39
FAILED: 0
IGNORED: 0

```

FIGURA 3.15. Ejecución de tests por consola.

Luego de realizar *commit* y *push* de los cambios locales, se pueden apreciar en la [figura 3.16](#) el listado de las versiones en [GitHub](#).

```

Test 'test_adc_service.c'
-----
Running test_adc_service.out...

Test 'test_display_service.c'
-----
Running test_display_service.out...

Test 'test_joystick_service.c'
-----
Running test_joystick_service.out...

Test 'test_measuring_services.c'
-----
Running test_measuring_services.out...

Test 'test_motors_service.c'
-----
Running test_motors_service.out...

Test 'test_robot_position_state.c'
-----
Running test_robot_position_state.out...

Test 'test_wifi_service.c'
-----
Running test_wifi_service.out...

-----
TEST OUTPUT
-----
[test motors service.c]
- "initializing mcpwm gpio..."
- "Configuring Initial Parameters of mcpwm..."
- "initializing mcpwm gpio..."
- "Configuring Initial Parameters of mcpwm..."

[test wifi service.c]
- "wifi init softap finished. SSID:1 password:1 channel:1"
- "station 12:34:56:78:9A:8C leave, AID=1"
- "station 12:34:56:78:9A:8C leave, AID=1"

-----
OVERALL TEST SUMMARY
-----
TESTED: 39
PASSED: 39
FAILED: 0
IGNORED: 0

```

FIGURA 3.15. Ejecución de tests por consola.

Luego de realizar *commit* y *push* de los cambios locales se pueden apreciar en la [imagen 3.16](#) el listado de las versiones en [Github](#).

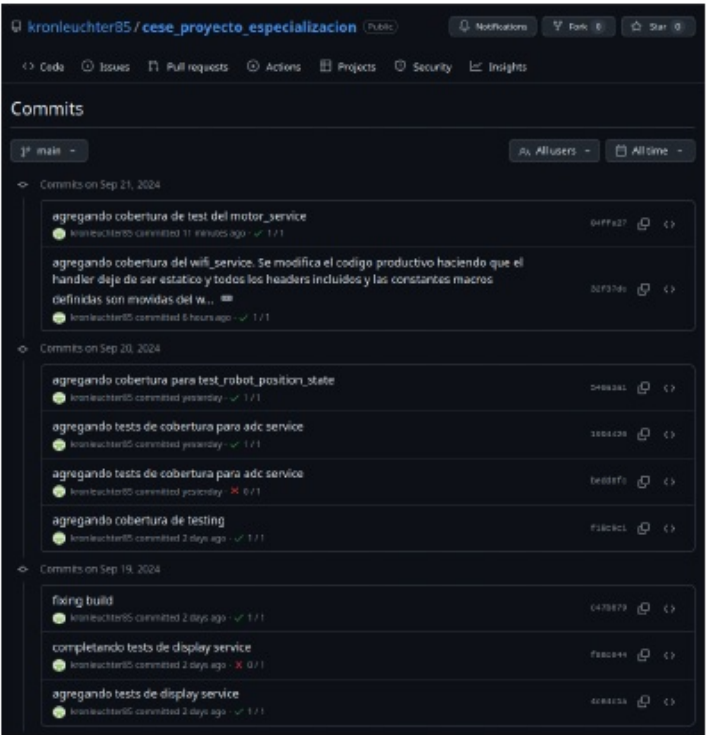


FIGURA 3.16. Listado de *commits* en Github.

En la [figura 3.17](#), se pueden apreciar los diferentes builds disparados en Cloud Build referenciando los commits de [GitHub](#).

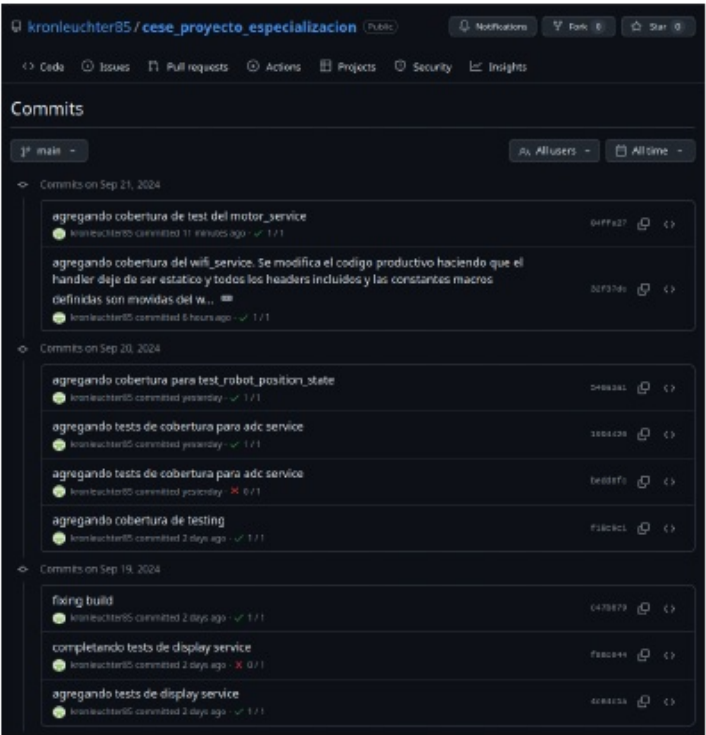


FIGURA 3.16. Listado de *commits* en Github.

En la [imagen 3.17](#) se pueden apreciar los diferentes builds disparados en Cloud Build referenciando los commits de [Github](#).

Status	Build	Source	Ref	Commit	Created	Duration
Success	8492ac78	konleachre@konleachre-projects-expected-ai	main	8492ac78	8/21/24, 11:43 AM	2 min 40 sec
Success	448eae83	konleachre@konleachre-projects-expected-ai	main	329370c3	8/21/24, 5:38 AM	3 min
Success	8f0e79c1	konleachre@konleachre-projects-expected-ai	main	346a3c12	8/20/24, 7:08 AM	2 min 40 sec
Success	1e229a4d	konleachre@konleachre-projects-expected-ai	main	1566a310	8/20/24, 7:30 AM	3 min 13 sec
Success	12b5e8b9	konleachre@konleachre-projects-expected-ai	main	3ed58c15	8/20/24, 7:23 AM	0 min 14 sec
Success	83c7c713	konleachre@konleachre-projects-expected-ai	main	118d9c12	8/19/24, 7:19 PM	2 min 18 sec
Success	15c49c75	konleachre@konleachre-projects-expected-ai	main	4f726b12	8/19/24, 6:18 AM	3 min 15 sec
Success	611ae05f	konleachre@konleachre-projects-expected-ai	main	506c0412	8/19/24, 6:13 AM	2 min 16 sec
Success	6534c12f	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/19/24, 6:03 AM	2 min 49 sec
Success	1a95e619	konleachre@konleachre-projects-expected-ai	main	4c84c1a10	8/19/24, 6:02 AM	32 sec
Success	8f04e41d	konleachre@konleachre-projects-expected-ai	main	1a6d1a12	8/19/24, 11:51 AM	2 min 26 sec
Success	c96ac999	konleachre@konleachre-projects-expected-ai	main	78412312	8/19/24, 9:27 AM	5 sec
Success	c76e9931	konleachre@konleachre-projects-expected-ai	main	20c02512	8/19/24, 9:25 AM	5 sec
Success	1970b179	konleachre@konleachre-projects-expected-ai	main	a18b0a12	8/19/24, 9:23 AM	6 sec
Success	11e19f1a	konleachre@konleachre-projects-expected-ai	main	1f6813210	8/19/24, 5:58 AM	6 sec
Success	c76e9931	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/17/24, 9:15 AM	2 min 16 sec
Success	c0093939	konleachre@konleachre-projects-expected-ai	main	16d99112	8/16/24, 4:48 PM	4 min 44 sec
Success	c7707709	konleachre@konleachre-projects-expected-ai	main	4d681112	8/16/24, 4:49 PM	2 min
Success	c95c84ba	konleachre@konleachre-projects-expected-ai	main	7239e112	8/16/24, 4:20 PM	2 min 21 sec
Success	c0086d13	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:29 PM	2 min 16 sec
Success	c76e9931	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:28 PM	6 sec
Success	c91b27a9	konleachre@konleachre-projects-expected-ai	main	c91b27a9	8/16/24, 2:25 PM	7 sec
Success	1b4d11e4	konleachre@konleachre-projects-expected-ai	main	52f60912	8/16/24, 2:23 PM	5 sec
Success	c7724d17	konleachre@konleachre-projects-expected-ai	main	36237d10	8/16/24, 2:28 PM	6 sec
Success	c0086d13	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:09 PM	2 min 23 sec
Success	c0086d13	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:08 PM	-
Success	c88c32ef	konleachre@konleachre-projects-expected-ai	main	81d19d12	8/16/24, 2:05 PM	5 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	82693712	8/16/24, 2:03 PM	10 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	2988a112	8/16/24, 2:02 PM	6 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	2c333d12	8/16/24, 1:55 PM	2 min 7 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 5:24 PM	5 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 5:14 PM	5 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:58 PM	2 min 14 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:58 PM	2 min 32 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:46 PM	2 min 12 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:40 PM	1 min 55 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:40 PM	5 sec

FIGURA 3.17. Listado de *builds* en Google CloudBuild.

Finalmente, en la figura 3.18 se pueden apreciar las imágenes Docker versionadas y almacenadas en Artifact Registry.

Status	Build	Source	Ref	Commit	Created	Duration
Success	8492ac78	konleachre@konleachre-projects-expected-ai	main	8492ac78	8/21/24, 11:43 AM	2 min 40 sec
Success	448eae83	konleachre@konleachre-projects-expected-ai	main	329370c3	8/21/24, 5:38 AM	3 min
Success	8f0e79c1	konleachre@konleachre-projects-expected-ai	main	346a3c12	8/20/24, 7:08 AM	2 min 40 sec
Success	1e229a4d	konleachre@konleachre-projects-expected-ai	main	1566a310	8/20/24, 7:30 AM	3 min 13 sec
Success	12b5e8b9	konleachre@konleachre-projects-expected-ai	main	3ed58c15	8/20/24, 7:23 AM	0 min 14 sec
Success	83c7c713	konleachre@konleachre-projects-expected-ai	main	118d9c12	8/19/24, 7:19 PM	2 min 18 sec
Success	15c49c75	konleachre@konleachre-projects-expected-ai	main	4f726b12	8/19/24, 6:18 AM	3 min 15 sec
Success	611ae05f	konleachre@konleachre-projects-expected-ai	main	506c0412	8/19/24, 6:13 AM	2 min 16 sec
Success	6534c12f	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/19/24, 6:03 AM	2 min 49 sec
Success	1a95e619	konleachre@konleachre-projects-expected-ai	main	4c84c1a10	8/19/24, 6:02 AM	32 sec
Success	8f04e41d	konleachre@konleachre-projects-expected-ai	main	1a6d1a12	8/19/24, 11:51 AM	2 min 26 sec
Success	c96ac999	konleachre@konleachre-projects-expected-ai	main	78412312	8/19/24, 9:27 AM	5 sec
Success	c76e9931	konleachre@konleachre-projects-expected-ai	main	20c02512	8/19/24, 9:25 AM	5 sec
Success	1970b179	konleachre@konleachre-projects-expected-ai	main	a18b0a12	8/19/24, 9:23 AM	6 sec
Success	11e19f1a	konleachre@konleachre-projects-expected-ai	main	1f6813210	8/19/24, 5:58 AM	6 sec
Success	c76e9931	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/17/24, 9:15 AM	2 min 16 sec
Success	c0093939	konleachre@konleachre-projects-expected-ai	main	16d99112	8/16/24, 4:48 PM	4 min 44 sec
Success	c7707709	konleachre@konleachre-projects-expected-ai	main	4d681112	8/16/24, 4:49 PM	2 min
Success	c95c84ba	konleachre@konleachre-projects-expected-ai	main	7239e112	8/16/24, 4:20 PM	2 min 21 sec
Success	c0086d13	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:29 PM	2 min 16 sec
Success	c76e9931	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:28 PM	6 sec
Success	c91b27a9	konleachre@konleachre-projects-expected-ai	main	c91b27a9	8/16/24, 2:25 PM	7 sec
Success	1b4d11e4	konleachre@konleachre-projects-expected-ai	main	52f60912	8/16/24, 2:23 PM	5 sec
Success	c7724d17	konleachre@konleachre-projects-expected-ai	main	36237d10	8/16/24, 2:28 PM	6 sec
Success	c0086d13	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:09 PM	2 min 23 sec
Success	c0086d13	konleachre@konleachre-projects-expected-ai	main	4d3dc1a10	8/16/24, 2:08 PM	-
Success	c88c32ef	konleachre@konleachre-projects-expected-ai	main	81d19d12	8/16/24, 2:05 PM	5 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	82693712	8/16/24, 2:03 PM	10 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	2988a112	8/16/24, 2:02 PM	6 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	2c333d12	8/16/24, 1:55 PM	2 min 7 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 5:24 PM	5 sec
Success	c49b4d11	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 5:14 PM	5 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:58 PM	2 min 14 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:58 PM	2 min 32 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:46 PM	2 min 12 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:40 PM	1 min 55 sec
Success	c7723d12	konleachre@konleachre-projects-expected-ai	main	c7723d12	8/15/24, 4:40 PM	5 sec

FIGURA 3.17. Listado de *builds* en Google CloudBuild.

Finalmente en la imagen 3.18 se pueden apreciar las imágenes Docker versionadas y almacenadas en Artifact Registry.

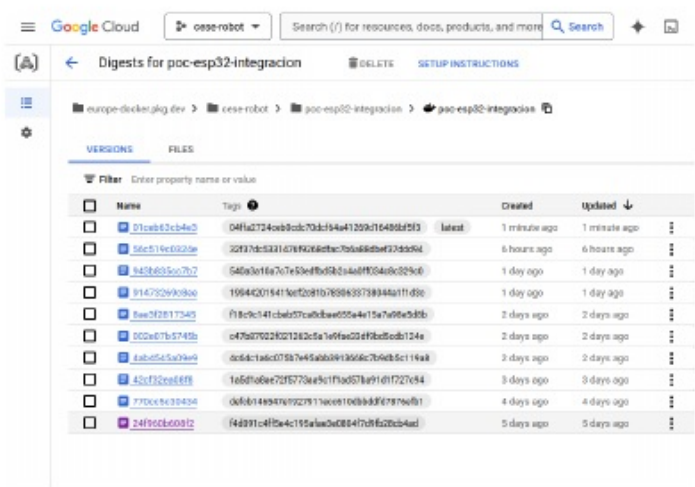


FIGURA 3.18. Listado de versiones de imágenes docker en Google ArtifactRegistry.

3.5. Reportes de ejecución y cobertura de testing unitario

A continuación, se presentan los reportes de testing generados por la herramienta *ceedling* con el complemento *gcov* donde se puede apreciar el nivel de cobertura logrado para cada servicio.

En la figura 3.19, se puede apreciar la salida por pantalla tras la ejecución local de *ceedling* con el plugin de cobertura, en donde se evidencia la cantidad de test cases.

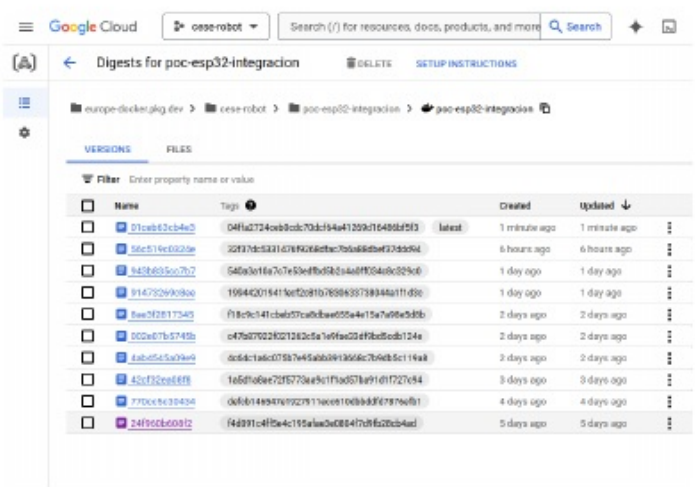


FIGURA 3.18. Listado de versiones de imágenes docker en Google ArtifactRegistry.

3.5. Reportes de ejecución y cobertura de testing unitario

A continuación se presentan los reportes de testing generados por la herramienta *ceedling* con el complemento *gcov* donde se puede apreciar el nivel de cobertura logrado para cada servicio.

En la imagen 3.19, se puede apreciar la salida por pantalla tras la ejecución local de *ceedling* con el plugin de cobertura, en donde se evidencia la cantidad de test cases.

```
-----
GCC: OVERALL TEST SUMMARY
-----
TESTED: 39
PASSED: 39
FAILED: 0
IGNORED: 0

-----
GCC: CODE COVERAGE SUMMARY
-----
adc_service.c Lines executed:100.00% of 17
adc_service.c No branches
adc_service.c Calls executed:100.00% of 13

display_service.c Lines executed:100.00% of 11
display_service.c No branches
display_service.c Calls executed:100.00% of 7

joystick_service.c Lines executed:91.67% of 24
joystick_service.c Branches executed:100.00% of 12
joystick_service.c Taken at least once:83.33% of 12
joystick_service.c Calls executed:100.00% of 2

measuring_services.c Lines executed:77.78% of 36
measuring_services.c Branches executed:100.00% of 10
measuring_services.c Taken at least once:88.88% of 18
measuring_services.c Calls executed:71.43% of 7

motors_service.c Lines executed:100.00% of 30
motors_service.c Branches executed:100.00% of 4
motors_service.c Taken at least once:75.00% of 4
motors_service.c Calls executed:100.00% of 15

robot_position_state.c Lines executed:81.48% of 27
robot_position_state.c Branches executed:100.00% of 24
robot_position_state.c Taken at least once:87.58% of 24
robot_position_state.c No calls

wifi_service.c Lines executed:100.00% of 19
wifi_service.c Branches executed:100.00% of 4
wifi_service.c Taken at least once:100.00% of 4
wifi_service.c Calls executed:100.00% of 15
```

FIGURA 3.19. Reportes de testing por consola.

En la figura 3.20, se pueden apreciar los detalles de la cobertura por cada servicio.

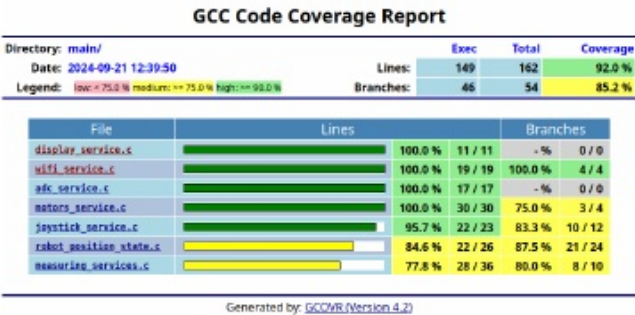


FIGURA 3.20. Reportes de testing web.

```
-----
GCC: OVERALL TEST SUMMARY
-----
TESTED: 39
PASSED: 39
FAILED: 0
IGNORED: 0

-----
GCC: CODE COVERAGE SUMMARY
-----
adc_service.c Lines executed:100.00% of 17
adc_service.c No branches
adc_service.c Calls executed:100.00% of 13

display_service.c Lines executed:100.00% of 11
display_service.c No branches
display_service.c Calls executed:100.00% of 7

joystick_service.c Lines executed:91.67% of 24
joystick_service.c Branches executed:100.00% of 12
joystick_service.c Taken at least once:83.33% of 12
joystick_service.c Calls executed:100.00% of 2

measuring_services.c Lines executed:77.78% of 36
measuring_services.c Branches executed:100.00% of 10
measuring_services.c Taken at least once:88.88% of 18
measuring_services.c Calls executed:71.43% of 7

motors_service.c Lines executed:100.00% of 30
motors_service.c Branches executed:100.00% of 4
motors_service.c Taken at least once:75.00% of 4
motors_service.c Calls executed:100.00% of 15

robot_position_state.c Lines executed:81.48% of 27
robot_position_state.c Branches executed:100.00% of 24
robot_position_state.c Taken at least once:87.58% of 24
robot_position_state.c No calls

wifi_service.c Lines executed:100.00% of 19
wifi_service.c Branches executed:100.00% of 4
wifi_service.c Taken at least once:100.00% of 4
wifi_service.c Calls executed:100.00% of 15
```

FIGURA 3.19. Reportes de testing por consola.

En la siguiente imagen 3.20 se pueden apreciar los detalles de la cobertura por cada servicio.

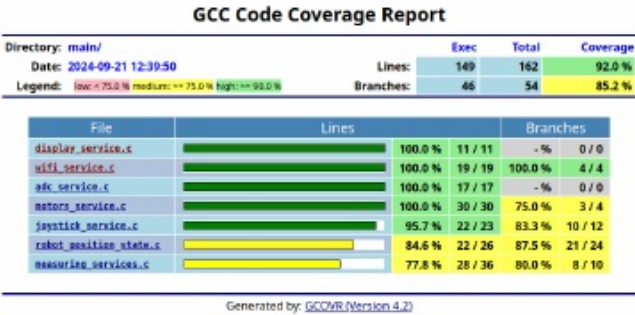


FIGURA 3.20. Reportes de testing web.

4.2.2. Verificación del módulo de control del display

Se verificó visualmente que el display mostraba los caracteres programados en la prueba de concepto con una intensidad de luz aceptable para poder leerlos apropiadamente. En las figuras 4.1, 4.8, 4.10 y 4.12 puede apreciarse el funcionamiento del módulo del display bajo diferentes condiciones de luminosidad.

4.2.3. Verificación del módulo de control de motores

Se verificó visualmente que individualmente el motor pudiera girar en ambos sentidos. Luego, al implementarse los cuatro motores con sus ruedas, se probó que se puedan realizar los giros en todas las direcciones. En los videos [53] y [54] puede apreciarse el funcionamiento del módulo de control de los motores.

4.2.4. Verificación del módulo de medición de temperatura y humedad

Se verificó visualmente que los valores obtenidos por el sensor DHT11 coincidieran con los esperados en relación a la temperatura en el interior del lugar de experimentación y que la humedad detectada se aproximara a los valores reportados por Google. En las figuras 4.2, 4.3, 4.4 y 4.5 puede apreciarse el funcionamiento del módulo de medición de temperatura y humedad.

4.2.5. Verificación del módulo de medición de presión atmosférica

Se verificó visualmente que el valor obtenido por el sensor BMP280 fuera cercano a lo esperado en relación al valor reportado por Google. En las figuras 4.7 y 4.7 puede apreciarse el funcionamiento del módulo de presión atmosférica.

4.2.6. Verificación del módulo de medición de luminosidad

Se verificó visualmente que los valores obtenidos del fotorresistor, una vez transformados a valores absolutos porcentuales, reflejaban el nivel de luminosidad ambiental del interior del lugar de experimentación. En las figuras 4.8, 4.10 y 4.12 puede apreciarse el funcionamiento del módulo de luminosidad.

4.2.7. Verificación del módulo de comunicación UTP sobre Wi-Fi

Por medio de dos programas UDP, uno cliente y uno servidor, se probó el establecimiento de la comunicación UDP entre dos ESP32. Posteriormente, se incorporó el servicio de comunicaciones UDP en el robot, y desde el programa cliente se enviaban las acciones que representaban las direcciones del movimiento (FORWARD, BACKWARD, LEFT, RIGHT). Se observó visualmente cómo el robot giraba sus ruedas en función de los comandos enviados. Finalmente, se incorporó el módulo de comunicaciones en el joystick y activó el desplazamiento en cada una de sus direcciones. El robot se desplazó correctamente en respuesta a cada comando recibido. En los videos [55], [56] y [57] puede apreciarse el funcionamiento de la comunicación UDP sobre Wi-Fi entre el robot y el joystick.

4.3. Pruebas funcionales y validación del producto

El proceso de validación y pruebas del producto, se realizó comparando el resultado obtenido con los valores esperados en el alcance del proyecto.

4.2.2. Verificación del módulo de control del display

Se verificó visualmente que el display mostraba los caracteres programados en la prueba de concepto con una intensidad de luz aceptable para poder leerlos apropiadamente. En las figuras 4.1, 4.8, 4.9 y 4.10 puede apreciarse el funcionamiento del módulo del display bajo diferentes condiciones de luminosidad.

4.2.3. Verificación del módulo de control de motores

Se verificó visualmente que individualmente el motor pudiera girar en ambos sentidos. Luego, al implementarse los cuatro motores con sus ruedas, se probó que se puedan realizar los giros en todas las direcciones. En los videos [53] y [54] puede apreciarse el funcionamiento del módulo de control de los motores.

4.2.4. Verificación del módulo de medición de temperatura y humedad

Se verificó visualmente que los valores obtenidos por el sensor DHT11 coincidieran con los esperados en relación a la temperatura en el interior del lugar de experimentación y que la humedad detectada se aproximara a los valores reportados por Google. En las figuras 4.7, 4.7, 4.7 y 4.7 puede apreciarse el funcionamiento del módulo de medición de temperatura y humedad.

4.2.5. Verificación del módulo de medición de presión atmosférica

Se verificó visualmente que el valor obtenido por el sensor BMP280 fuera cercano a lo esperado en relación al valor reportado por Google. En las figuras 4.7 y 4.7 puede apreciarse el funcionamiento del módulo de presión atmosférica.

4.2.6. Verificación del módulo de medición de luminosidad

Se verificó visualmente que los valores obtenidos del fotorresistor, una vez transformados a valores absolutos porcentuales, reflejaban el nivel de luminosidad ambiental del interior del lugar de experimentación. En las figuras 4.8, 4.9 y 4.10 puede apreciarse el funcionamiento del módulo de luminosidad.

4.2.7. Verificación del módulo de comunicación UTP sobre Wi-Fi

Por medio de dos programas UDP, uno cliente y uno servidor, se probó el establecimiento de la comunicación UDP entre dos ESP32. Posteriormente, se incorporó el servicio de comunicaciones UDP en el robot, y desde el programa cliente se enviaban las acciones que representaban las direcciones del movimiento (FORWARD, BACKWARD, LEFT, RIGHT). Se observó visualmente cómo el robot giraba sus ruedas en función de los comandos enviados. Finalmente, se incorporó el módulo de comunicaciones en el joystick y activó el desplazamiento en cada una de sus direcciones. El robot se desplazó correctamente en respuesta a cada comando recibido. En los videos [55], [56] y [57] puede apreciarse el funcionamiento de la comunicación UDP sobre Wi-Fi entre el robot y el joystick.

4.3. Pruebas funcionales y validación del producto

El proceso de validación y pruebas del producto, se realizó comparando el resultado obtenido con los valores esperados en el alcance del proyecto.



FIGURA 4.6. Medición de presión atmosférica en el interior.

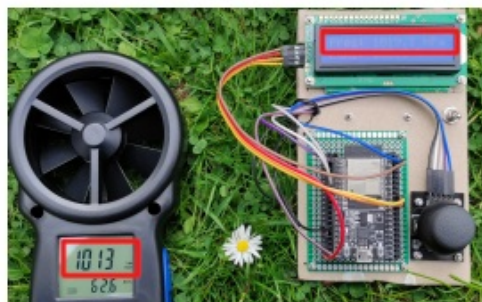


FIGURA 4.7. Medición de presión atmosférica en el exterior.

En la siguiente sección puede encontrarse el video que muestra el funcionamiento del módulo de mediciones, en el que se aprecia el funcionamiento del módulo de medición de presión atmosférica [58].

4.3.4. Prueba y validación del módulo de medición de luminosidad ambiental

Se compararon los valores medidos por el módulo de medición de luminosidad en diferentes escenarios:

- En exteriores durante el día con luz ambiental.
- En interiores con luz ambiental.
- En interiores a oscuras.

Los resultados mostraron que los valores porcentuales indicados por el módulo de medición de luminosidad son consistentes con los niveles de luz detectados por el ojo humano. Además, con el fin de cuantificar en unidades estandares el proceso de validación, se comparó los niveles porcentuales reportados por el robot con los niveles de iluminancia medidos en Lux [60], utilizando la aplicación Light Meter [61] instalada en un smartphone Android.

En las figuras 4.8, 4.9, 4.10, 4.11, 4.12 y 4.13 pueden apreciarse los resultados de las mediciones.



FIGURA 4.6. Medición de presión atmosférica en el interior.

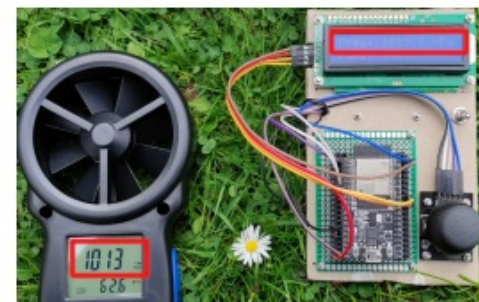


FIGURA 4.7. Medición de presión atmosférica en el exterior.

En la siguiente sección puede encontrarse el video que muestra el funcionamiento del módulo de mediciones, en el que se aprecia el funcionamiento del módulo de medición de presión atmosférica [58].

4.3.4. Prueba y validación del módulo de medición de luminosidad ambiental

Se compararon los valores medidos por el módulo de medición de luminosidad basado en un fotorresistor percibidos por el ojo humano sin utilizar ningún dispositivo de medición. Se realizó la medición en diferentes escenarios:

- En exteriores durante el día con luz ambiental.
- En interiores con luz ambiental.
- En interiores a oscuras.

Los resultados mostraron que los valores porcentuales indicados por el módulo de medición de luminosidad son consistentes con los niveles de luz detectados por el ojo humano. En las figuras 4.8, 4.9 y 4.10 pueden apreciarse los resultados de las mediciones.



FIGURA 4.8. Medición de luminosidad ambiental en el exterior durante el día.

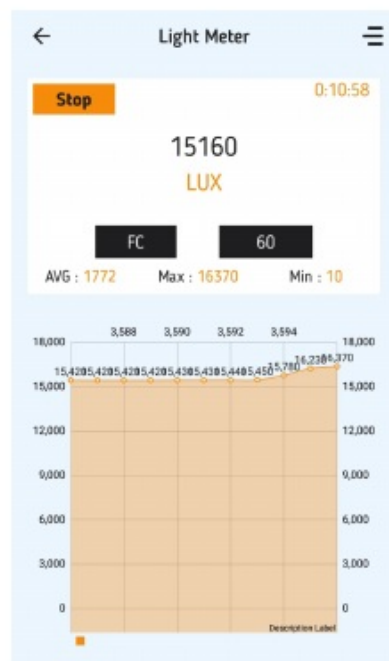


FIGURA 4.9. Iluminancia [Lux] reportada por la aplicación Light Meter en el exterior durante el día.



FIGURA 4.8. Medición de luminosidad ambiental en el exterior durante el día.



FIGURA 4.9. Medición de luminosidad ambiental en interiores durante el día.



FIGURA 4.10. Medición de luminosidad ambiental en interiores durante la noche.

En la siguiente sección puede encontrarse el video que muestra el funcionamiento del módulo de mediciones, en el que se aprecia el funcionamiento del módulo de medición de luminosidad ambiental [58].

4.3.5. Prueba y validación del control y desplazamiento del robot

Se verificó el control del desplazamiento del robot de forma visual por medio de accionar el joystick en las diferentes coordenadas (X;Y) y se controló que:



FIGURA 4.10. Medición de luminosidad ambiental en interiores durante el día.

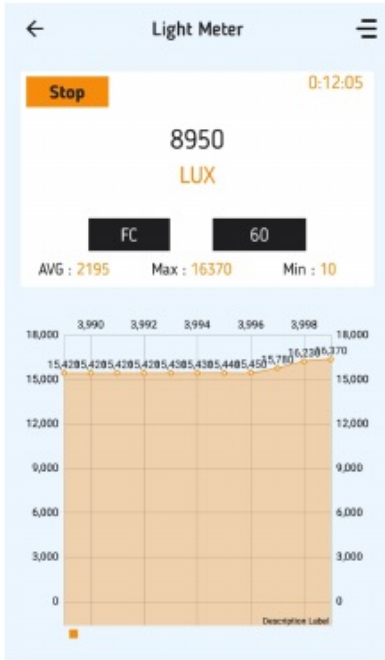


FIGURA 4.11. Iluminancia [Lux] reportada por la aplicación Light Meter en interiores durante el día.

- La dirección del movimiento del robot sea acorde al accionamiento del joystick.
- El tiempo de respuesta en el movimiento del robot y tras el accionar del joystick sea mínimo, permitiendo una buena experiencia de usuario.

En la siguiente sección pueden encontrarse los videos [60] y [61] evidenciando la demostración de este experimento.

4.4. Videos del producto durante el ensamblado y experimentación

En las siguientes subsecciones se listan los videos realizados durante el proceso de demostración del producto funcionando así como los grabados casualmente durante armado y prototipado del mismo.

4.4.1. Videos demostrativos del producto final

Los experimentos realizados para evidenciar el cumplimiento con los requerimientos funcionales del producto son los siguientes:

- Demo - Hardware del producto [62].
- Demo - Comunicación Wi-Fi [55].
- Demo - Control de movimiento de las ruedas [60].
- Demo - Medición y visualización de parámetros ambientales [58].
- Demo - Control de desplazamiento en un circuito [61].
- Demo - Visualización del Display en la oscuridad [59].

4.4.2. Videos durante el prototipado y ensamblado del robot

- Prototipado Robot v1 - Ensamblado (1) [53].
- Prototipado Robot v1 - Ensamblado (2) [54].
- Prototipado Robot v1 - Ensamblado (3) [63].
- Prototipado Robot v1 - Ensamblado (4) [64].
- Prototipado Robot v2 - Comunicación Joystick Robot (1) [56].
- Prototipado Robot v2 - Comunicación Joystick Robot (2) [57].
- Prototipado Desplazamiento (alimentación USB) [65].
- Prototipado Desplazamiento (alimentación por pilas) [66].

4.5. Documentación del producto

Se desarrolló la documentación del producto compuesta de los siguientes entregables

- Documentación técnica [67].



FIGURA 4.12. Medición de luminosidad ambiental en interiores durante la noche.



FIGURA 4.13. Iluminancia [Lux] reportada por la aplicación Light Meter en interiores durante la noche.

En la siguiente sección puede encontrarse el video que muestra el funcionamiento del módulo de mediciones, en el que se aprecia el funcionamiento del módulo de medición de luminosidad ambiental [58].

- Manual de usuario [68].

4.3.5. Prueba y validación del control y desplazamiento del robot

Se verificó el control del desplazamiento del robot de forma visual por medio de accionar el joystick en las diferentes coordenadas (X;Y) y se controló que:

- La dirección del movimiento del robot sea acorde al accionamiento del joystick.
- El tiempo de respuesta en el movimiento del robot y tras el accionar del joystick sea mínimo, permitiendo una buena experiencia de usuario.

En la siguiente sección pueden encontrarse los videos [62] y [63] evidenciando la demostración de este experimento.

4.4. Videos del producto durante el ensamblado y experimentación

En las siguientes subsecciones se listan los videos realizados durante el proceso de demostración del producto funcionando así como los grabados casualmente durante armado y prototipado del mismo.

4.4.1. Videos demostrativos del producto final

Los experimentos realizados para evidenciar el cumplimiento con los requerimientos funcionales del producto son los siguientes:

- Demo - Hardware del producto [64].
- Demo - Comunicación Wi-Fi [55].
- Demo - Control de movimiento de las ruedas [62].
- Demo - Medición y visualización de parámetros ambientales [58].
- Demo - Control de desplazamiento en un circuito [63].
- Demo - Visualización del Display en la oscuridad [59].

4.4.2. Videos durante el prototipado y ensamblado del robot

- Prototipado Robot v1 - Ensamblado (1) [53].
- Prototipado Robot v1 - Ensamblado (2) [54].
- Prototipado Robot v1 - Ensamblado (3) [65].
- Prototipado Robot v1 - Ensamblado (4) [66].
- Prototipado Robot v2 - Comunicación Joystick Robot (1) [56].
- Prototipado Robot v2 - Comunicación Joystick Robot (2) [57].
- Prototipado Desplazamiento (alimentación USB) [67].
- Prototipado Desplazamiento (alimentación por pilas) [68].

4.5. Documentación del producto

Se desarrolló la documentación del producto compuesta de los siguientes entregables

- Documentación técnica [69].
- Manual de usuario [70].

Capítulo 5

Conclusiones

Conclusiones del trabajo...

- ¿Cuál es el grado de cumplimiento de los requerimientos?
- ¿Cuán fielmente se pudo seguir la planificación original (cronograma incluido)?
- ¿Se manifestó algunos de los riesgos identificados en la planificación? ¿Fue efectivo el plan de mitigación? ¿Se debió aplicar alguna otra acción no contemplada previamente?
- Si se debieron hacer modificaciones a lo planificado ¿Cuáles fueron las causas y los efectos?
- ¿Qué técnicas resultaron útiles para el desarrollo del proyecto y cuáles no tanto?

5.1. Próximos pasos

Acá se indica cómo se podría continuar el trabajo más adelante.

Capítulo 5

Conclusiones

Conclusiones del trabajo...

- ¿Cuál es el grado de cumplimiento de los requerimientos?
- ¿Cuán fielmente se pudo seguir la planificación original (cronograma incluido)?
- ¿Se manifestó algunos de los riesgos identificados en la planificación? ¿Fue efectivo el plan de mitigación? ¿Se debió aplicar alguna otra acción no contemplada previamente?
- Si se debieron hacer modificaciones a lo planificado ¿Cuáles fueron las causas y los efectos?
- ¿Qué técnicas resultaron útiles para el desarrollo del proyecto y cuáles no tanto?

5.1. Próximos pasos

Acá se indica cómo se podría continuar el trabajo más adelante.

Bibliografía

- [1] Latam Mining. *Robots y minería: Gobierno argentino quiere implementarlos*. URL: <https://www.latam-mining.com/robots-y-mineria-gobierno-argentino-quiere-implementarlos/>.
- [2] Diario de Cuyo. *Gobierno pone la mira en el desarrollo de robots para la actividad minera*. URL: <https://www.diariodecuyo.com.ar/politica/Gobierno-pone-la-mira-en-el-desarrollo-de-robots-para-la-actividad-minera-20200202-0052.html>.
- [3] Universidad Nacional de San Juan. *Robots en la minería*. URL: http://www.unsj.edu.ar/home/noticias_detalle/4810/1.
- [4] Ing. Nelson Dario García Hurtado e Ing. Melvin Andrés González Pino. *Robot de exploración terrestre Geobot*. URL: https://www.unipamplona.edu.co/unipamplona/portallG/home_40/recursos/01_general/revista_1/09102011/v01_09.pdf.
- [5] Ing. Hernán L. Helguero Velásquez1 e Ing. Rubén Medinaceli Tórrez. *Robot Minero: Sistema Detector de Gases utilizando Sensores en Tiempo Real MIN – SIS 1.0 SDG-STR*. URL: http://www.scielo.org.bo/scielo.php?script=sci_arttext&pid=S2519-53522020000100003.
- [6] Boston Dynamics. *Spot*. URL: <https://www.bostondynamics.com/products/spot>.
- [7] Waygate Technologies. *BIKE - An advanced crawler robot for remote visual inspection*. URL: <https://www.bakerhughes.com/waygate-technologies/robotic-inspection/bike>.
- [8] Espressif. *ESP32*. URL: <https://www.espressif.com/en/products/socs/esp32>.
- [9] Espressif. *ESP32-WROOM-32D Datasheet*. URL: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32d_esp32-wroom-32u_datasheet_en.pdf.
- [10] Mouser. *DHT11 datasheet*. URL: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>.
- [11] Bosch. *BMP280 datasheet*. URL: <https://cdn-shop.adafruit.com/datasheets/BST-BMP280-DS001-11.pdf>.
- [12] Handson Technology. *PS2 Joy Stick for Arduino/Raspberry*. URL: <http://www.handsontec.com/dataspecs/accessory/PS2-Joystick.pdf>.
- [13] Espressif. *Analog to Digital Converter (ADC)*. URL: <https://docs.espressif.com/projects/esp-idf/en/v4.4/esp32/api-reference/peripherals/adc.html>.
- [14] Handson Technology. *I2C Serial Interface 1602 LCD Module*. URL: http://www.handsontec.com/dataspecs/module/I2C_1602_LCD.pdf.
- [15] Adafruit. *DC Gearbox Motor - TT Motor -200RPM - 3 to 6VDC*. URL: https://media.digikey.com/pdf/Data%20Sheets/Adafruit%20PDFs/3777_Web.pdf.

- [16] Handson Technology. *L298N Driver Module Datasheet*. URL: <https://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf>.
- [17] EEMB. *Li-Ion batteries 18650 3000 mAh*. URL: <http://www.kosmodrom.com.ua/pdf/LIR18650-3000mah.pdf>.
- [18] Farnell. *Li-Ion batteries AA 2600 mAh*. URL: <https://www.farnell.com/datasheets/3195148.pdf>.
- [19] AOPUTTRIVER. *Anemómetro digital AOPUTTRIVER AP-007-WM*. URL: <https://manuals.plus/m/a30ffaa3ac8fd2cf06cb5559777c3af166bcaeed18d1ee49a56ea6f10c66dd4b.pdf>.
- [20] CMake. *CMake*. URL: <https://cmake.org/>.
- [21] CMake. *Ninja*. URL: <https://cmake.org/cmake/help/latest/generator/Ninja.html>.
- [22] Espressif. *ESP-IDF Programming Guide 1 Get Started*. URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/>.
- [23] Readthedocs by Ruslan V. Uss. *ESP-IDF Components library*. URL: <https://esp-idf-lib.readthedocs.io/en/latest/>.
- [24] Espressif Programing Guide. *ESP-IDF Get Started*. URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/index.html>.
- [25] Docker. *Docker*. URL: <https://docker.com/>.
- [26] ThrowTheSwitch. *ThrowTheSwitch - Ceedling*. URL: <https://github.com/ThrowTheSwitch/Ceedling>.
- [27] ThrowTheSwitch. *ThrowTheSwitch - CMock*. URL: <https://github.com/ThrowTheSwitch/CMock>.
- [28] ThrowTheSwitch. *ThrowTheSwitch - Unity Test*. URL: <https://github.com/ThrowTheSwitch/Unity>.
- [29] ThrowTheSwitch. *ThrowTheSwitch - Ceedling/GCov*. URL: <https://github.com/ThrowTheSwitch/Ceedling/blob/master/plugins/gcov/README.md>.
- [30] Github. *Github*. URL: <https://github.com/>.
- [31] Google Cloud Platform. *Google Cloud Build*. URL: <https://cloud.google.com/build>.
- [32] Google Cloud Platform. *Google Artifact Registry*. URL: <https://cloud.google.com/artifact-registry>.
- [33] Visualstudio. *Visualstudio Code*. URL: <https://code.visualstudio.com/>.
- [34] Ubuntu. *Ubuntu*. URL: <https://ubuntu.com/>.
- [35] Espressif. *ESP-IDF WiFi*. URL: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-guides/wifi.html>.
- [36] Espressif. *ESP-IDF - Wi-Fi SoftAP Example*. URL: https://github.com/espressif/esp-idf/tree/v4.4/examples/wifi/getting_started/softAP.
- [37] Gonzalo Carreno. *POC ESP32-WiFi v4.4*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-wifi-ap-v4.4.
- [38] Espressif. *ESP-IDF ADC1 Example*. URL: <https://github.com/espressif/esp-idf/tree/v4.0.3/examples/peripherals/adc>.
- [39] Gonzalo Carreno. *POC ESP32-joystick*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-joystick.

Bibliografía

- [1] Latam Mining. *Robots y minería: Gobierno argentino quiere implementarlos*. URL: <https://www.latam-mining.com/robots-y-mineria-gobierno-argentino-quiere-implementarlos/>.
- [2] Diario de Cuyo. *Gobierno pone la mira en el desarrollo de robots para la actividad minera*. URL: <https://www.diariodecuyo.com.ar/politica/Gobierno-pone-la-mira-en-el-desarrollo-de-robots-para-la-actividad-minera-20200202-0052.html>.
- [3] Universidad Nacional de San Juan. *Robots en la minería*. URL: http://www.unsj.edu.ar/home/noticias_detalle/4810/1.
- [4] Ing. Nelson Dario García Hurtado e Ing. Melvin Andrés González Pino. *Robot de exploración terrestre Geobot*. URL: https://www.unipamplona.edu.co/unipamplona/portallig/home_40/recursos/01_general/revista_1/09102011/v01_09.pdf.
- [5] Ing. Hernán L. Helguero Velásquez1 e Ing. Rubén Medinaceli Tórrez. *Robot Minero: Sistema Detector de Gases utilizando Sensores en Tiempo Real MIN – SIS 1.0 SDG-STR*. URL: http://www.scielo.org.bo/scielo.php?script=sci_arttext&pid=S2519-53522020000100003.
- [6] Boston Dynamics. *Spot*. URL: <https://www.bostondynamics.com/products/spot>.
- [7] Waygate Technologies. *BIKE - An advanced crawler robot for remote visual inspection*. URL: <https://www.bakerhughes.com/waygate-technologies/robotic-inspection/bike>.
- [8] Espressif. *ESP32*. URL: <https://www.espressif.com/en/products/socs/esp32>.
- [9] Espressif. *ESP32-WROOM-32D Datasheet*. URL: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32d_esp32-wroom-32u_datasheet_en.pdf.
- [10] Mouser. *DHT11 datasheet*. URL: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>.
- [11] Bosch. *BMP280 datasheet*. URL: <https://cdn-shop.adafruit.com/datasheets/BST-BMP280-DS001-11.pdf>.
- [12] Handson Technology. *PS2 Joy Stick for Arduino/Raspberry*. URL: <http://www.handsontec.com/dataspecs/accessory/PS2-Joystick.pdf>.
- [13] Espressif. *Analog to Digital Converter (ADC)*. URL: <https://docs.espressif.com/projects/esp-idf/en/v4.4/esp32/api-reference/peripherals/adc.html>.
- [14] Handson Technology. *I2C Serial Interface 1602 LCD Module*. URL: http://www.handsontec.com/dataspecs/module/I2C_1602_LCD.pdf.
- [15] Adafruit. *DC Gearbox Motor - TT Motor -200RPM - 3 to 6VDC*. URL: https://media.digikey.com/pdf/Data%20Sheets/Adafruit%20PDFs/3777_Web.pdf.

- [40] Gonzalo Carreno. *POC ESP32-photoresistor*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-photoresistor.
- [41] UncleRus. *ESP32 - Example for dht driver*. URL: <https://github.com/UncleRus/esp-idf-lib/tree/master/examples/dht/default>.
- [42] Gonzalo Carreno. *POC ESP32-DHT11*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-dht11.
- [43] UncleRus. *ESP32 - Example for bmp280 driver*. URL: <https://github.com/UncleRus/esp-idf-lib/tree/master/examples/bmp280/default>.
- [44] Gonzalo Carreno. *POC ESP32-BMP280*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-bmp280.
- [45] Espressif. *L298N Dual H-Bridge Motor Driver*. URL: <https://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf>.
- [46] Espressif. *ESP-IDF Motor Control Pulse Width Modulator (MCPWM)*. URL: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/peripherals/mcpwm.html>.
- [47] Espressif. *MCPWM*. URL: <https://docs.espressif.com/projects/esp-idf/en/v4.2/esp32/api-reference/peripherals/mcpwm.html>.
- [48] Espressif. *MCPWM brushed dc motor control Example*. URL: https://github.com/espressif/esp-idf/tree/v4.2/examples/peripherals/mcpwm/mcpwm_brushed_dc_control.
- [49] Gonzalo Carreno. *POC ESP32-motor-pwm*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-motor-pwm.
- [50] ESP32 Tutoriales. *ESP32 I2C LCD with ESP-IDF*. URL: <https://esp32tutorials.com/i2c-lcd-esp32-esp-idf/>.
- [51] Gonzalo Carreno. *POC ESP32-DHT11*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-display.
- [52] Espressif. *Espressif Docker Image*. URL: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-guides/tools/idf-docker-image.html>.
- [53] Gonzalo Carreno. *Robot exploracion ambiental - Prototipado Ensamblado Robot v1 (1)*. URL: <https://youtu.be/tDXT1CsObWE>.
- [54] Gonzalo Carreno. *Robot exploracion ambiental - Prototipado Ensamblado Robot v1 (2)*. URL: <https://youtube.com/shorts/uGqJn2K0LbI>.
- [55] Gonzalo Carreno. *Robot de exploracion ambiental - Demo - Comunicacion WiFi*. URL: <https://youtu.be/CcBgvoKjLB0>.
- [56] Gonzalo Carreno. *Robot de exploracion ambiental - Prototipado Comunicacion Joystick Robot (1)*. URL: <https://youtu.be/SnRf6HSya88>.
- [57] Gonzalo Carreno. *Robot de exploracion ambiental - Prototipado Comunicacion Joystick Robot (2)*. URL: <https://youtu.be/jiisheyu95w>.
- [58] Gonzalo Carreno. *Robot de exploracion ambiental - Demo Medicion y visualizacion de parametros ambientales*. URL: https://youtu.be/BBOP3n8_gBg.
- [59] Gonzalo Carreno. *Robot de exploracion ambiental - Demo Visualizacion del Display en la oscuridad*. URL: <https://youtu.be/LwfYaotAi64>.

- [16] Handson Technology. *L298N Driver Module Datasheet*. URL: <https://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf>.
- [17] EEMB. *Li-Ion batteries 18650 3000 mAh*. URL: <http://www.kosmodrom.com.ua/pdf/LIR18650-3000mah.pdf>.
- [18] Farnell. *Li-Ion batteries AA 2600 mAh*. URL: <https://www.farnell.com/datasheets/3195148.pdf>.
- [19] AOPUTTRIVER. *Anemómetro digital AOPUTTRIVER AP-007-WM*. URL: <https://manuals.plus/m/a30ffaa3ac8fd2cf06cb5559777c3af166bcaeed18d1ee49a56ea6f10c66dd4b.pdf>.
- [20] CMake. *CMake*. URL: <https://cmake.org/>.
- [21] CMake. *Ninja*. URL: <https://cmake.org/cmake/help/latest/generator/Ninja.html>.
- [22] Espressif. *ESP-IDF Programming Guide | Get Started*. URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/>.
- [23] Readthedocs by Ruslan V. Uss. *ESP-IDF Components library*. URL: <https://esp-idf-lib.readthedocs.io/en/latest/>.
- [24] Espressif Programing Guide. *ESP-IDF Get Started*. URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/index.html>.
- [25] Docker. *Docker*. URL: <https://docker.com/>.
- [26] ThrowTheSwitch. *ThrowTheSwitch - Ceedling*. URL: <https://github.com/ThrowTheSwitch/Ceedling>.
- [27] ThrowTheSwitch. *ThrowTheSwitch - CMock*. URL: <https://github.com/ThrowTheSwitch/CMock>.
- [28] ThrowTheSwitch. *ThrowTheSwitch - Unity Test*. URL: <https://github.com/ThrowTheSwitch/Unity>.
- [29] ThrowTheSwitch. *ThrowTheSwitch - Ceedling/GCov*. URL: <https://github.com/ThrowTheSwitch/Ceedling/blob/master/plugins/gcov/README.md>.
- [30] Github. *Github*. URL: <https://github.com/>.
- [31] Google Cloud Platform. *Google Cloud Build*. URL: <https://cloud.google.com/build>.
- [32] Google Cloud Platform. *Google Artifact Registry*. URL: <https://cloud.google.com/artifact-registry>.
- [33] Visualstudio. *Visualstudio Code*. URL: <https://code.visualstudio.com/>.
- [34] Ubuntu. *Ubuntu*. URL: <https://ubuntu.com/>.
- [35] Espressif. *ESP-IDF WiFi*. URL: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-guides/wifi.html>.
- [36] Espressif. *ESP-IDF - Wi-Fi SoftAP Example*. URL: https://github.com/espressif/esp-idf/tree/v4.4/examples/wifi/getting_started/softAP.
- [37] Gonzalo Carreno. *POC ESP32-WiFi v4.4*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-wifi-ap-v4.4.
- [38] Espressif. *ESP-IDF ADC1 Example*. URL: <https://github.com/espressif/esp-idf/tree/v4.0.3/examples/peripherals/adc>.
- [39] Gonzalo Carreno. *POC ESP32-joystick*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/tree/main/pocs/esp32-joystick.

- [60] Gonzalo Carreno. *Robot de exploracion ambiental - Demo Control de movimiento de las ruedas*. URL: <https://youtu.be/FKXWx4Rqr7I>.
- [61] Gonzalo Carreno. *Robot de exploracion ambiental - Demo Control de desplazamiento en un circuito*. URL: <https://youtu.be/sosSGwCTyaY>.
- [62] Gonzalo Carreno. *Robot de exploracion ambiental - Demo Hardware*. URL: <https://youtu.be/RNBnDawVj6c>.
- [63] Gonzalo Carreno. *Robot exploracion ambiental - Prototipado Ensamblado Robot v1 (3)*. URL: <https://youtu.be/w9IOoE-d9Cw>.
- [64] Gonzalo Carreno. *Robot exploracion ambiental - Prototipado Ensamblado Robot v1 (4)*. URL: https://youtu.be/obkj-wM_wNU.
- [65] Gonzalo Carreno. *Robot de exploracion ambiental - Prototipado Desplazamiento (alimentacion USB)*. URL: https://youtu.be/_w8qdNWC-DQ.
- [66] Gonzalo Carreno. *Robot de exploracion ambiental - Prototipado Desplazamiento (alimentacion por pilas)*. URL: <https://youtu.be/-MxMXKzztHU>.
- [67] Gonzalo Carreno. *Robot de exploracion ambiental - Documentacion Tecnica*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/blob/main/docs/Documentacion-Tecnica.pdf.
- [68] Gonzalo Carreno. *Robot de exploracion ambiental - Manual de usuario*. URL: https://github.com/kronleuchter85/cese_proyecto_especializacion/blob/main/docs/Manual-De-Usuario-vFinal.pdf.