

Sprawozdanie z rozwoju projektu / dokumentacja

0. Changelog (od 27.03.2024)

27.03.2024

- dodano identyfikator CID w diagramie 7.1. Urządzenie monitoringu, przy komunikacji TCP identyfikuje się teraz przy użyciu CID (Camera identifier), zamiast przy użyciu adresu MAC (który w prawdzie jest zapisywany na serwerze, ale niewykorzystywany w celu identyfikacji urządzenia podczas połączeń TCP).
- zaktualizowano opis diagramu 7.1.
- dodano diagram 7.2. z opisem
- dodano diagram 7.3.

3.04.2024

- Poprawiono diagram 7.1.

10.04.2024

- Dodano tabelę charakterystyk GATT dla BLE (9.1.1)

17.04.2024

- Dodano diagram zadań i kolejek RTOS (9.2.1)

25.04.2024

- Zaktualizowano kolektywny diagram sekwencji dla przypadków użycia powiązanych z kooperacją urządzeń podczas rejestracji urządzenia (7.1)
- Update 7.2 and 7.3 diagrams with CKEY (camera authentication key)

9.05.2024

- Dodano TCP+UDP protocol docs

0.1 Todo

25.04.2024 - 9.2.2, 9.3, zaktualizować diagram klas dla urządzenia monitoringu

1. Proponowane technologie

Oprogramowanie serwerowe - Node.js + baza danych Sqlite

Aplikacja mobilna - JDK/Android

Oprogramowanie ESP32 - ESP-IDF (Espressif) / C

2. Przydział funkcji do osób / podział zespołu na realizowane moduły

A.S. - Amadeusz Sitnicki

A. R. - Adam Rosiak

B. C. - Bartłomiej Czerwiński

P. R. - Pavel Radkevich

J.K. - Jędrzej Kostyk

Symbol	Imię, Nazwisko	Podprojekt	Funkcja/Moduł	Ogólny opis / cel
#1-E1	A.S.	ESP32	CAMAU = Camera capture + Analyser + UDP	Implementacja kooperacji składającej się z 1. operacji niezbędnych do przechwytywania obrazu z sensora

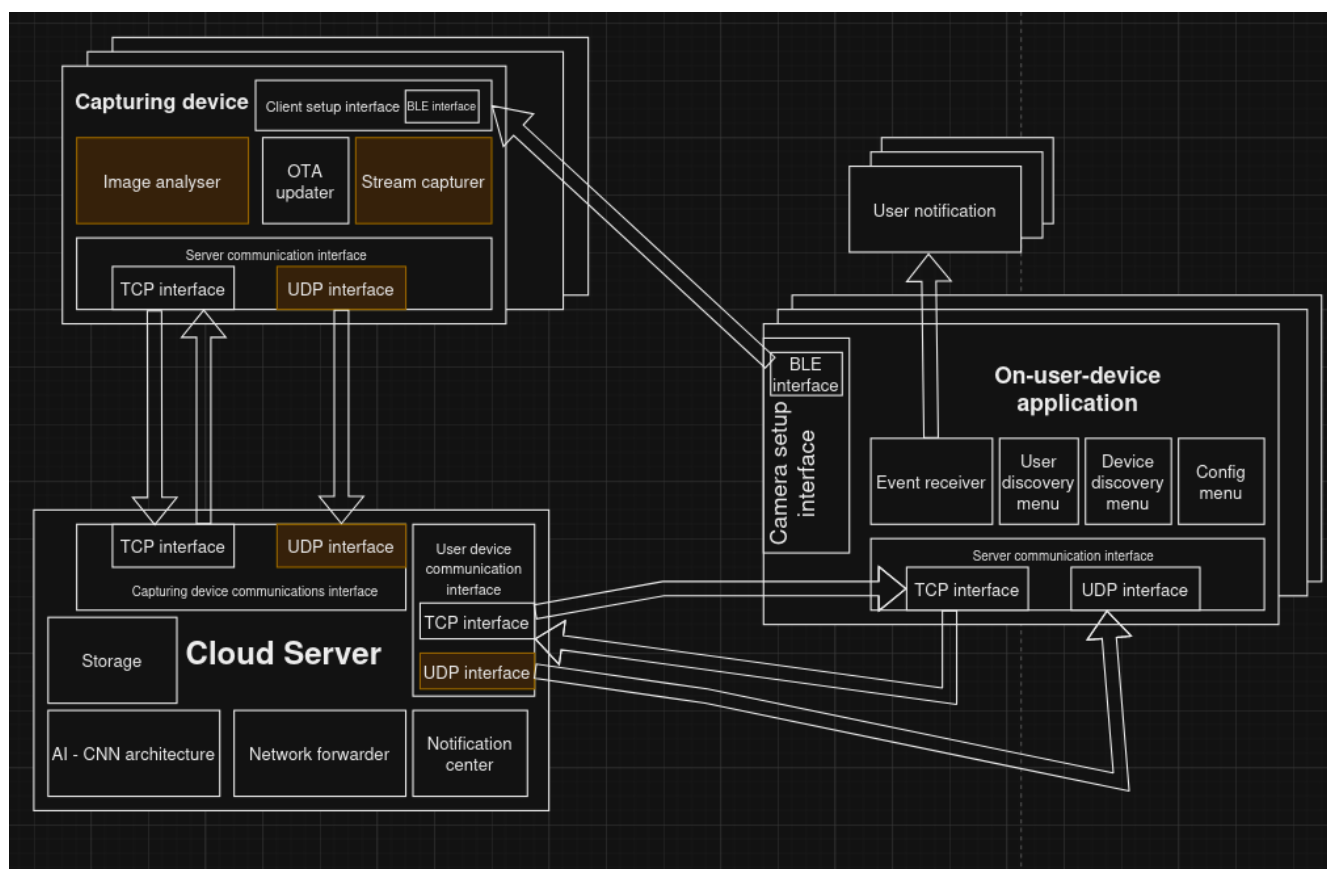
				<p>CCD, 2. operacji umożliwiających analizę obrazu z użyciem zasobooszczędnego algorytmu sztucznej inteligencji w celu wykrywania obecności człowieka na przechwyconym obrazie,</p> <p>3. operacji umożliwiających transmisję danych JFIF do serwera z wykorzystaniem komunikacji opartej na UDP</p>
#2-E2	A.S.	ESP32	OTA updater	Konfiguracja tablicy partycji oraz implementacja operacji umożliwiających zdalną modyfikację oprogramowania urządzenia
#3-E3	A.S.	ESP32	TCP client	Implementacja operacji, których celem jest zagwarantowanie, że oprogramowanie w prawidłowy sposób zareaguje na żądania otrzymywane ze strony serwera, np. żądanie transmisji obrazu niezależnie od wyniku klasyfikacji algorytmem AI, otrzymanie od serwera zmienionych parametrów obrazu, otrzymanie od serwera harmonogramu trybu zużycia energii, a także wysyłanie na serwer żądań, np. żądanie analizy danych JFIF ze strumienia UDP
#4-E4	A.S.	ESP32	Configuration manager	Implementacja operacji, które mają na celu wdrożenie nowej konfiguracji parametrów pracy urządzenia
#5-E5	A.S.	ESP32	BLE peripheral	Implementacja operacji, których celem aktywacja oraz dezaktywacja interfejsu Bluetooth Low Energy, umożliwienie odebrania danych dostępowych do sieci WLAN przesłanych przez aplikację mobilną
#6-S1	P.R.	Serwer	Moduł powiadomień	Wysyła powiadomienia do różnych instancji aplikacji mobilnej, gdy obecność człowieka na obrazie została potwierdzona przez serwerowy moduł AI, możliwe wykorzystanie SSE (text/event-stream)
#7-S2	A.S., J.K.	Serwer	Moduł klasyfikatora	Potwierdza lub odrzuca zgłoszony obraz przesłany w żądaniu analizy z

			obrazów	ESP32 na serwer, w zależności czy model AI stwierdził na obrazie wysokie prawdopodobieństwo obecności człowieka
#8-S3	A.R.	Serwer	Video persistence	Implementacja operacji zapewniających utrwalenie danych wideo w pamięci nieulotnej na serwerze i zapewnienie interfejsu dostępu do zapisanych plików z podziałem na różne urządzenia monitoringu
#9-S4	A.R.	Serwer	TCP server - esp32 communication submodule	Implementacja operacji, których celem jest zapewnienie prawidłowej komunikacji z urządzeniami monitoringu z wykorzystaniem protokołu TCP, zagwarantowanie, że serwer wyśle żądania do urządzenia monitorującego, np. żądanie analizy danych JFIF otrzymanych za pośrednictwem strumienia UDP lub żądanie ciągłej transmisji obrazu niezależnie od wyniku klasyfikacji wykonanej przez model AI, wysyłanie do urządzenia zmienionych parametrów obrazu, wysyłanie do urządzenia harmonogramu trybu zużycia energii, otrzymanie statusu urządzenia, informacji o bieżącej konfiguracji. Dodatkowo przeprowadzanie aktualizacji OTA
#10-S5	A.R.	Serwer	UDP server - esp32 communication submodule	Implementacja operacji odbierających transmisję obrazu zarówno w przypadku gdy miało miejsce żądanie analizy jak i w przypadku transmisji niezależnie od wyniku klasyfikacji algorytmem AI
#11-S6	P.R., J.K.	Serwer	TCP/UDP server - mobile application communication submodule	Transmisja zapisanych w pamięci serwera lub odbieranych z urządzeń monitoringu strumieni UDP do instancji aplikacji mobilnej, możliwe wykorzystanie istniejących protokołów wyższej warstwy, np. RTSP/RTMP. Implementacja operacji umożliwiających odbiór od aplikacji mobilnej żądania transmisji obrazu niezależnie od wyniku klasyfikacji

				<p>algorytmem AI, otrzymanie od aplikacji mobilnej konfiguracji parametrów obrazu, otrzymanie harmonogramu trybu zużycia energii, a także odbiór ze strony aplikacji żądań takich jak utworzenie nowego użytkownika, zalogowanie/wylogowanie, przekazywanie danych związanych z kontem użytkownika oraz wysłanie do aplikacji informacji o pomyślnym zarejestrowaniu się urządzenia monitorującego w systemie</p>
#12-S7	P.R., J.K.	Serwer	Users & devices management	<p>Moduł zarządza bazą danych i umieszcza w niej informacje związane z kontami użytkowników, odpowiada za zarządzanie sesją i kontem użytkownika, umieszcza w bazie danych informacje o urządzeniach, których właścicielem jest użytkownik, udostępnia abstrakcję użytkownika i urządzenia dla pozostałych modułów</p>
		Serwer	Moduł logiki- interafejs esp32 -- interfejs AM	<p>Zapewnia synchronizację zdarzeń między podmodułami kontroli- komunikacji sieciowej (aplikacja- mobilna, esp32)</p>
#13-A1	B.C.	Aplikacja mobilna	Klient TCP - moduł kontroli urządzenia i użytkownika	<p>Implementacja operacji, których celem jest zapewnienie prawidłowej komunikacji z serwerem z wykorzystaniem protokołu TCP, zagwarantowanie, że aplikacja wyśle do serwera na prośbę użytkownika np. żądanie ciągłej transmisji obrazu niezależnie od wyniku klasyfikacji wykonanej przez model AI (podgląd obrazu na żywo), wysyłanie do serwera zmienionych parametrów obrazu, wysyłanie do serwera harmonogramu trybu zużycia energii dla określonego urządzenia użytkownika, odbiór od serwera statusu urządzenia, odbiór listy dostępnych wersji oprogramowania, wymiana danych związanych z kontem i sesją użytkownika z</p>

				serwerem
#14-A2	B.C.	Aplikacja mobilna	Klient TCP/UDP - moduł odbioru transmisji wideo	Operacje umożliwiające odbiór transmisji wideo za pośrednictwem serwera. Preferowany klient RTSP/RTMP o ile serwer będzie wykorzystywał ten sam protokół
#15-A3	B.C.	Aplikacja mobilna	BLE Central	Zaimplementowanie operacji, których celem jest obsługa interfejsu Bluetooth Low Energy, umożliwienie wysłania danych dostępowych do sieci WLAN - przesył zainicjowany przez aplikację mobilną do urządzenia monitoringu, które użytkownik pragnie zarejestrować, odbiór informacji czy urządzenie podłączyło się do sieci pomyślnie (jest to jedyny przypadek bezpośredniej komunikacji aplikacja mobilna - urządzenie monitorujące)
#16-A4	B.C.	Aplikacja mobilna	TCP client / Notifications handling submodule (HTTP/SSE)	Zaimplementowanie operacji realizujących czynności podejmowane przez aplikację w przypadku otrzymania od serwera informacji o wykryciu człowieka na obrazie poprzez protokół TCP / protokół wyższej warstwy - preferowane HTTP SSE (MIME text/event-stream), wyświetlanie powiadomień
#17-A5	B.C.	Aplikacja mobilna	Interfejs graficzny użytkownika	Przeglądarka zapisanych na serwerze klipów wideo, wyświetlanie transmisji na żywo, integracja z pozostałymi modułami - GUI panelu użytkownika, rejestracja/logowanie użytkownika, rejestracja urządzeń (w tym podawanie danych dostępowych do sieci) i listy urządzeń, panel konfiguracji, panel aktualizacji wersji oprogramowania, panel harmonogramowania trybów zużycia energii

3. Poglądowy diagram ogólnej architektury systemu

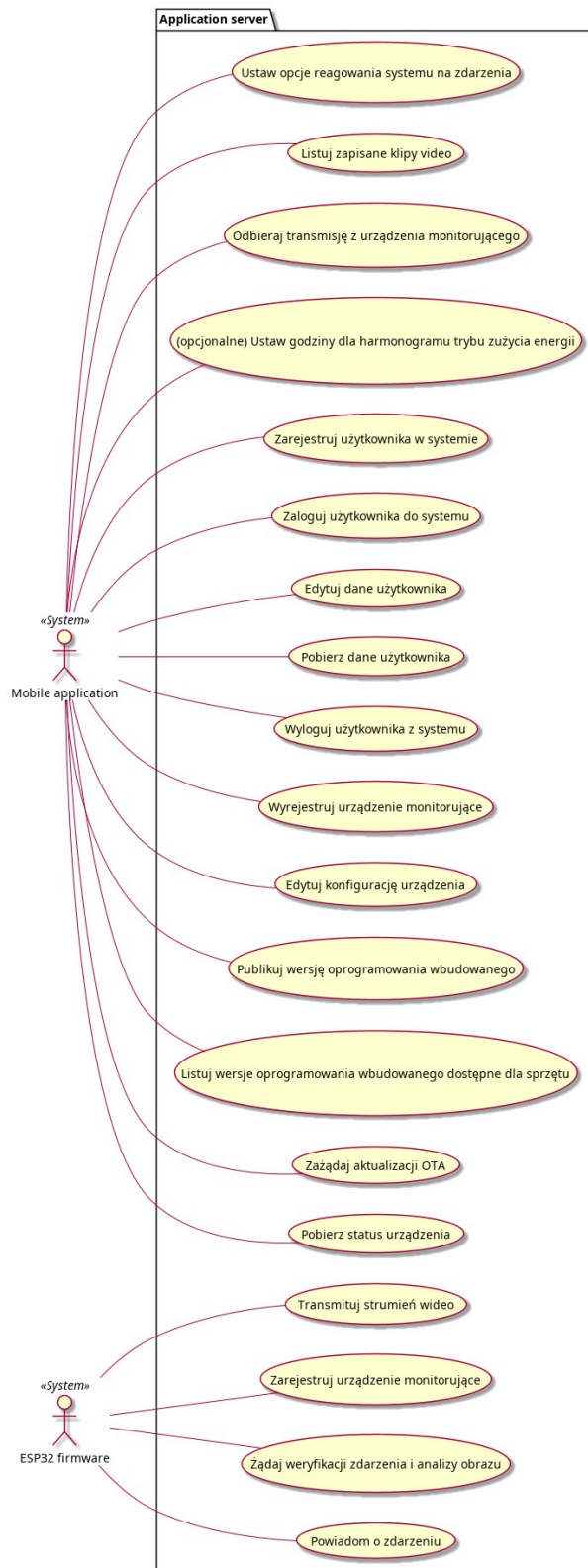


4. Diagramy przypadków użycia

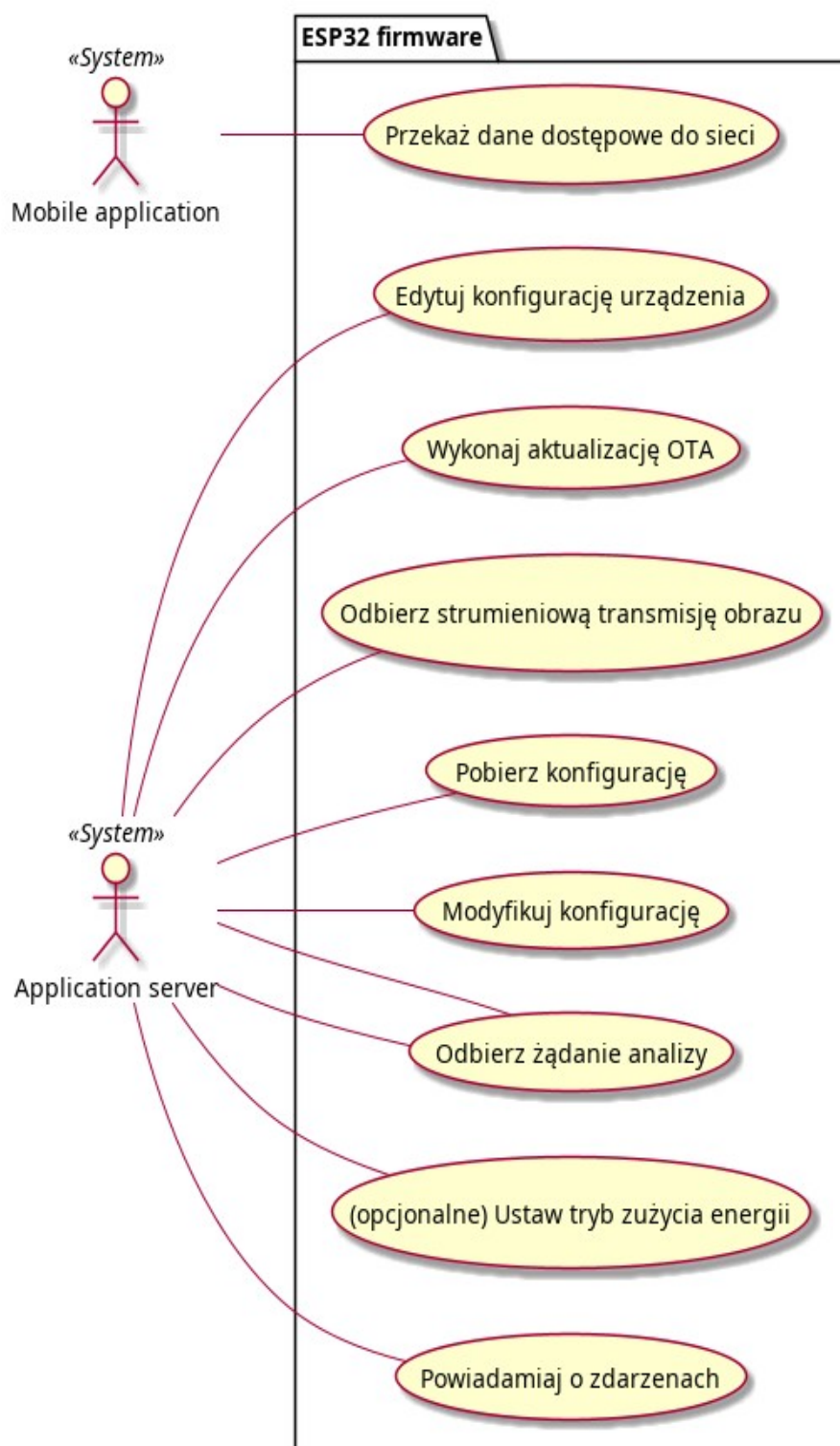
4.1. Diagram przypadków użycia - Aplikacja mobilna



4.2. Diagram przypadków użycia - Serwer



4.3. Diagram przypadków użycia - ESP32



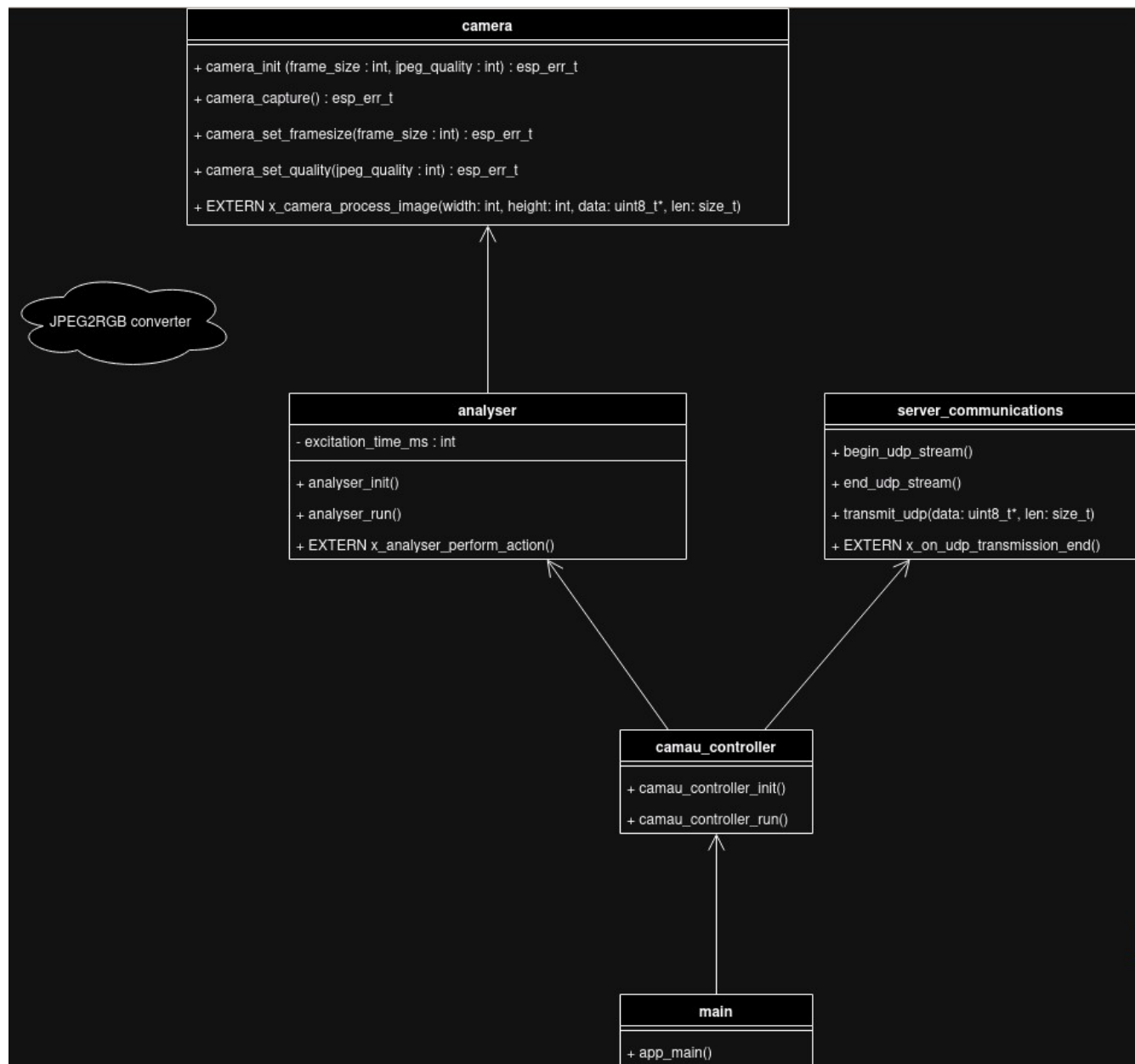
5. Spis przypadków użycia i modułów zaangażowanych w dostarczenie przypadku użycia i przetworzenie danych z nim związanych

Symbol przyp. uż.	Węzeł sprzętowy implementacji	Symbol modułu	Aktor wymagający p.u.	Nazwa przypadku użycia	Moduł po drugiej stronie interfejsu	Uwagi
P1	ESP32	E5	Aplikacja mobilna	Przełącz dane dostępne do sieci		
P2		E4	Serwer	Edytuj konfigurację urządzenia		
P3		E2		Wykonaj aktualizację OTA		
P4		E1, E3		Odbierz transmisję obrazu		
P5		E3, E4		Pobierz konfigurację		
P6		E3, E4		Modyfikuj konfigurację		
P7		E1, E3		Odbierz żądanie analizy		
P8		E3, E4		(opcjonalne) Ustaw tryb zużycia energii		
P9		E1, E3		Powiadamiaj o zdarzeniach		
P10	Serwer	S6, S7	Aplikacja mobilna	Zarejestruj użytkownika w systemie		
P11		S6, S7		Zaloguj użytkownika do systemu		
P12		S6, S7		Edytuj dane użytkownika		
P13		S6, S7		Pobierz dane użytkownika		
P14		S6, S7		Wyloguj użytkownika z systemu		
P15		S6, S7		Wyrejestruj urządzenie monitorujące		
P16		S4, S6, S7		Edytuj konfigurację urządzenia		
P18		S6, S7		Listuj wersje oprogramowania wbudowanego dostępne dla sprzętu		
P19		S4, S6		Zażądaj aktualizacji OTA		
P20		S4, S6		Pobierz status urządzenia		
P21		S1, S2		Ustaw opcje reagowania systemu na zdarzenia		
P22		S3, S6		Listuj zapisane klipy video		
P23		S2, S3, S5		Odbieraj transmisję z urządzenia monitorującego		
P24		S4, S6		(opcjonalne) Ustaw godziny dla harmonogramu trybu zużycia energii		
P26		S1, S2	ESP32	Powiadom o zdarzeniu		
P27		S2, S3		Transmituj strumień wideo		
P28		S2, S4		Żądaj weryfikacji zdarzenia i analizy obrazu		
P29		S4, S7		Zarejestruj urządzenie monitorujące		
P30	Aplik. mobilna	A1, A5	Użytkownik	Zarejestruj użytkownika w systemie		
P31		A1, A5		Zaloguj użytkownika do systemu		
P32		A1, A5		Edytuj dane użytkownika		
P33		A1, A5		Wyświetl profil użytkownika		

P34		A1, A5		Wyloguj użytkownika z systemu		
P35		A1, A3, A5		Zarejestruj urządzenie monitorujące		
P36		A1, A5		Wyrejestruj urządzenie monitorujące		
P37		A1, A5		Przeprowadź podstawową konfigurację urządzenia		
P39		A1, A5		Wyświetl wersje oprogramowania wbudowanego dostępne dla sprzętu		
P40		A1, A5		Zdalnie aktualizuj firmware		
P41		A1, A5		Wyświetl status urządzenia		
P42		A4		Wyświetl powiadomienie		
P43		A1, A5		Edytuj ustawienia reagowania systemu na zdarzenia		
P44		A2, A5		Przeglądaj zapisane klipy video		
P45		A2, A5		Oglądaj transmisję z urządzenia monitorującego		
P46		A1, A5		(opcjonalne) Zarządzaj godzinami harmonogramu trybu zużycia energii		
P47		A4	Serwer	Powiadom o zdarzeniu		
P48		A2		Transmituj strumień wideo		
P49		A3	ESP32	Pobierz dane dostępne do sieci		
P50	Serwer	?	?	Publikuj wersję oprogramowania wbudowanego		Express ?

6. Diagramy klas

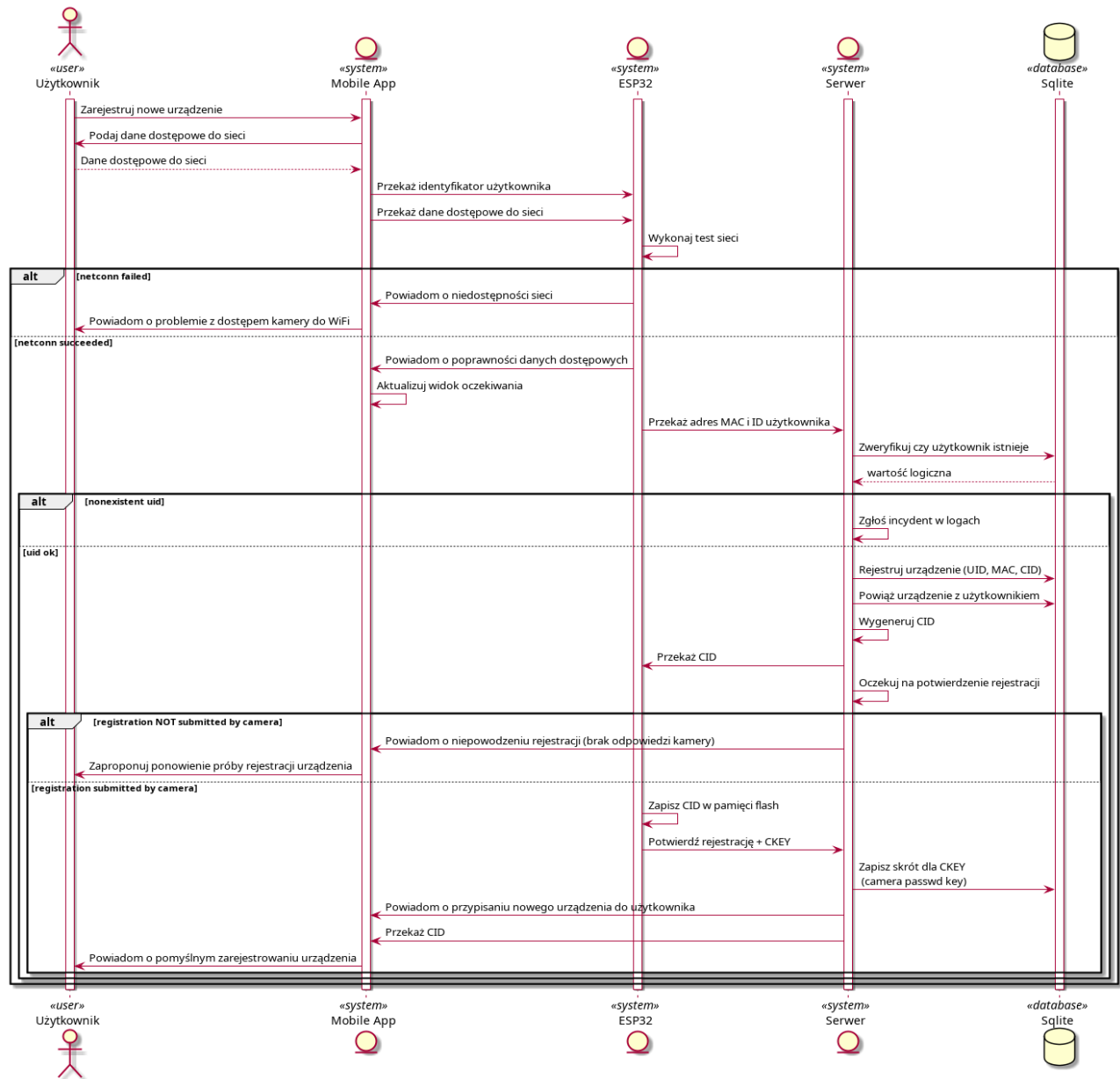
6.1. Diagram klas - E1 [niekompletny] (ostatnia aktualizacja - 12.03.2024).



7. Scenariusze i diagramy interakcji

7.1. Kolektywny diagram sekwencji dla przypadków użycia powiązanych z kooperacją urządzeń podczas rejestracji urządzenia - przekrój systemu informatycznego

- Aplikacja mobilna, gdy użytkownik chce: **zarejestrować urządzenie monitorujące (P35)**
- Aplikacja mobilna, gdy ESP32 chce: **pobrać dane dostępne do sieci (P49)**
- Serwer, gdy ESP32 chce : **zarejestrować urządzenie monitorujące (P29)**
- ESP32, gdy aplikacja mobilna chce : **przekazać dane dostępne do sieci (P1)**

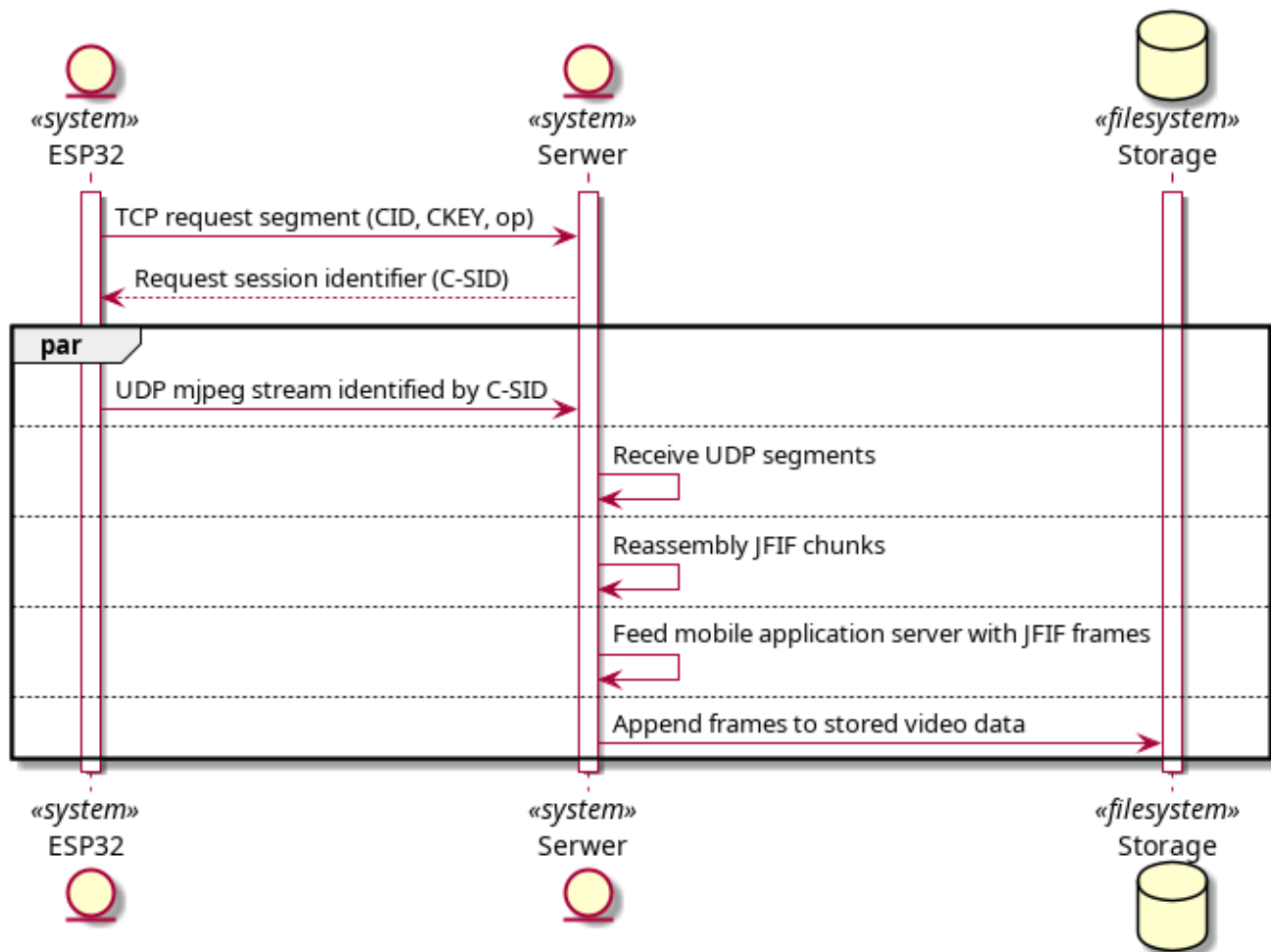


UID- user identifier

CID - camera identifier - stored in ESP32's non-volatile memory as the result of registration

CKEY - camera passwd key - generated by ESP32, stored in its memory, used for camera authentication

7.2. Diagram sekwencji dla przypadków użycia (P4, P7, P23 - transmisja obrazu)



CID - camera identifier permanently stored on ESP32 since device registration

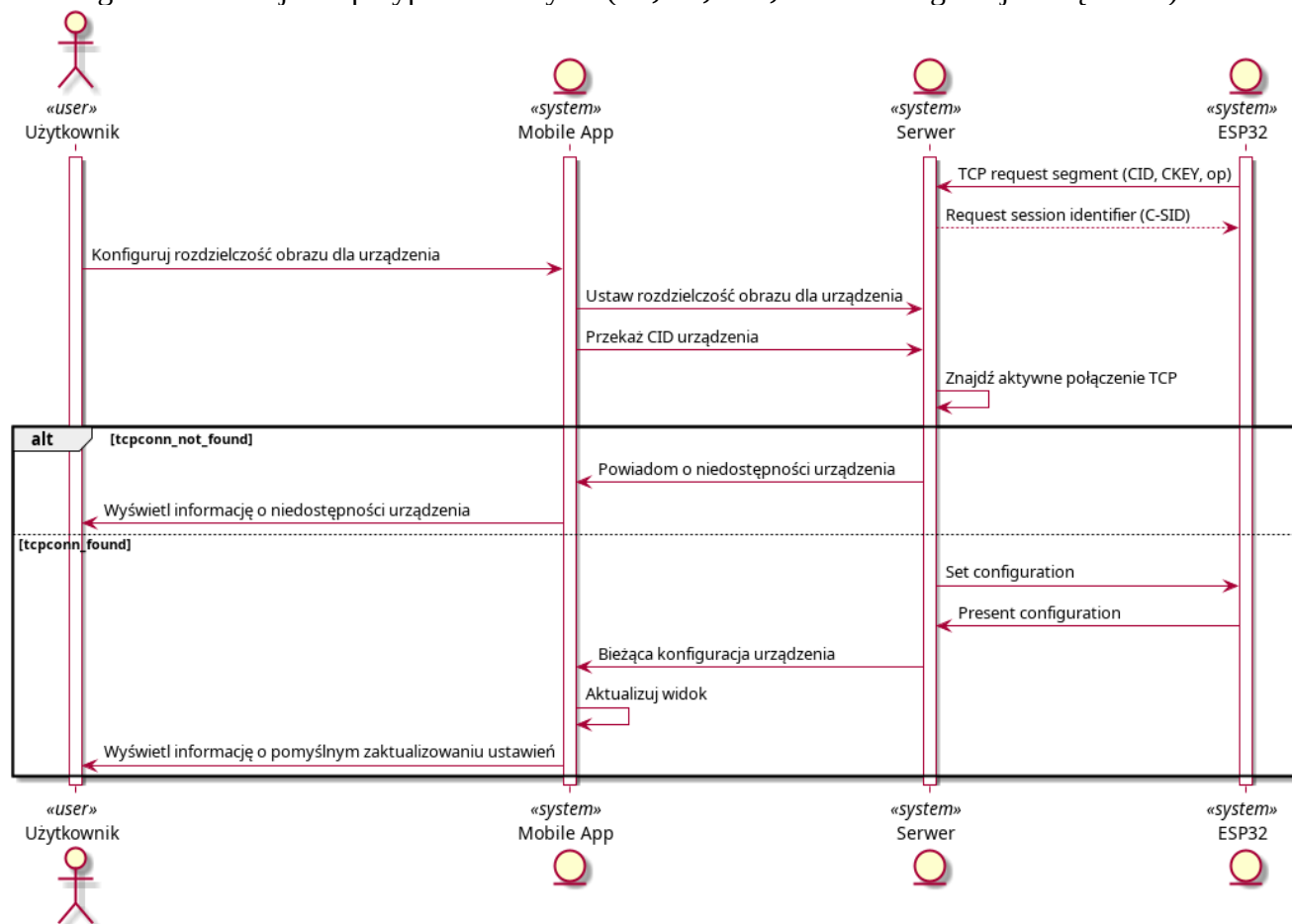
op - operation code which determines what the camera device wants to do

C-SID - camera session identifier - can be used by the ESP32 while UDP streaming, so that the server can link the C-SID with a specific UDP session and specific camera CID.

JFIF chunks are reassembled by storing UDP segments in a priority queue based on IP packet index.

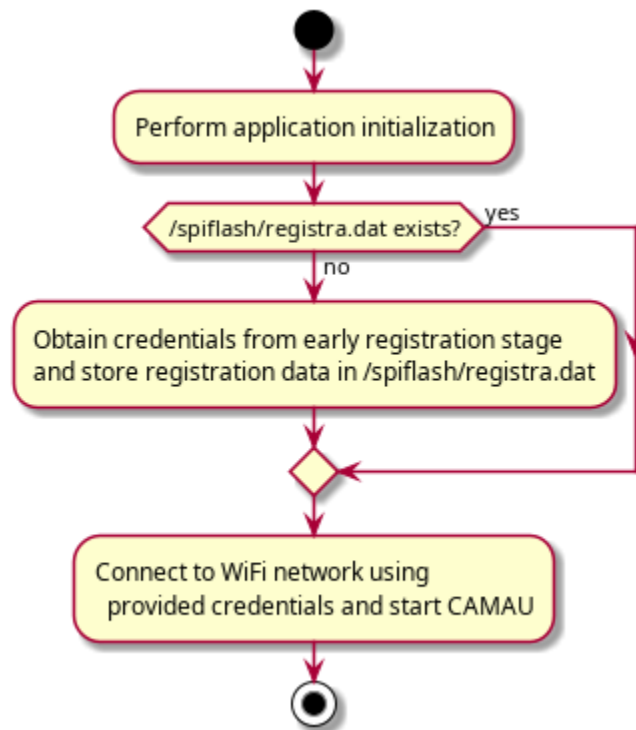
Server parallel tasks are recommended to be synchronized using Producer-Consumer queues. This is similar to the UDP interface receiving side prototype written using python (see the UDP_central.py script placed in the ESP32 subproject as an example).

7.3. Diagram sekwencji dla przypadków użycia (P5, P6, P16, P37 - konfiguracja urządzenia)



8. Diagramy czynności

8.1. Wysokopoziomowy diagram czynności dla rejestracji urządzenia monitorującego (ESP32)



9. Interfejsy komunikacyjne

9.1 Komunikacja podczas rejestracji między aplikacją mobilną a urządzeniem monitoringu - BLE (dodatkowy opis w załączniku "ble_interface.pdf")

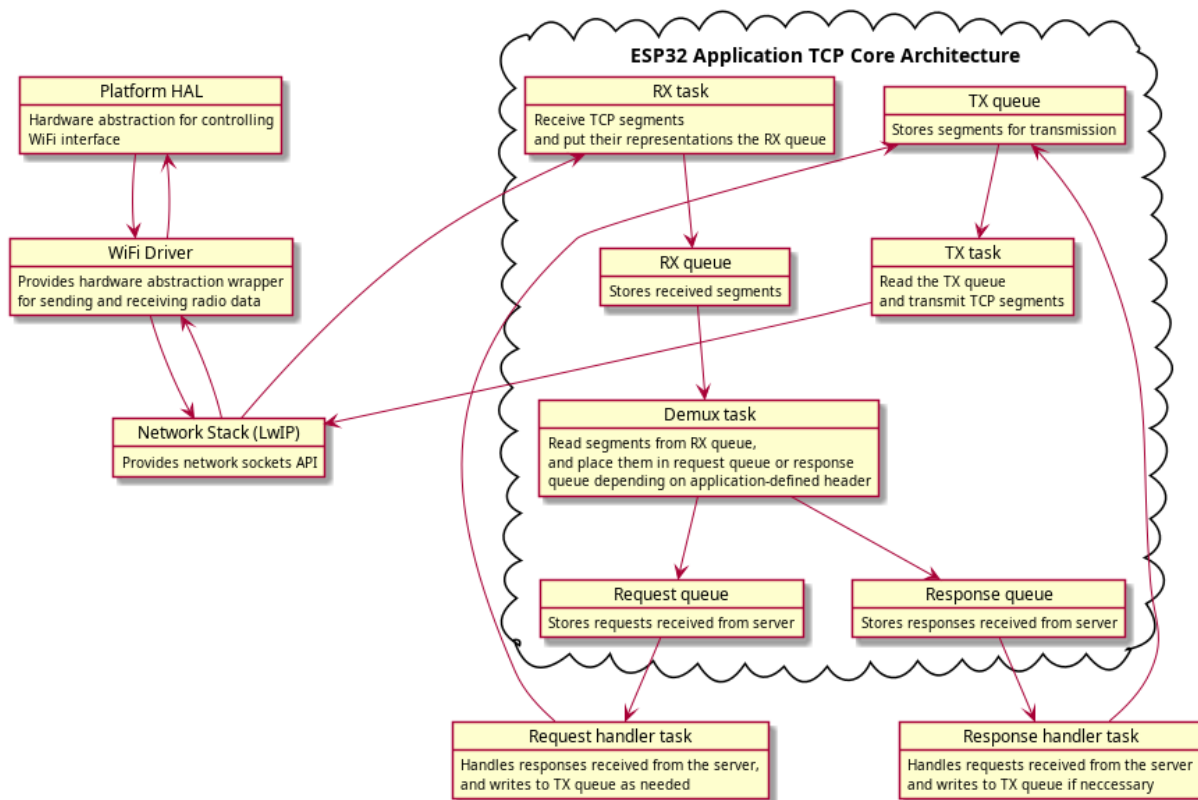
9.1.1 Tabela charakterystyk GATT

UUID	char value handle	Uprawnienia/char props	Znaczenie w kontekście warstwy aplikacji	Wartość
c4cdddf0-bcf3-1a85-2348-bea2aa91c9d8	0x000c	write (ch. prop=0x08)	Nazwa sieci wifi_ssid	Ciąg znaków o długości w przedziale [1; 20]
4e44eb46-	0x000e	write (ch.	Hasło do	Ciąg znaków

UUID	char value handle	Uprawnienia/ char props	Znaczenie w kontekście warstwy aplikacji	Wartość
7080-9d83-ac4a-e9f1b82b13d8		prop=0x08)	sieci wifi_psk	o długości w przedziale [1; 40]
b8b40435-be4f-55a5-a447-9c14ddcd20fc	0x0010	write (ch. prop=0x08)	Identyfikator użytkownika user_id	Ciąg znaków o długości 16
dd0a6b3f-86d8-68a9-4141-3ed99c4c2ac7	0x0012	notify (ch. prop=0x10)	Stan połączenia z siecią WiFi network_stat e	NETWORK_STAT E_WIFI_DISCON NECTED = 0x00 lub NETWORK_STAT E_WIFI_CONNEC TED=0x01

9.2 Komunikacja kontroli przepływu zdarzeń w systemie informatycznym - między serwerem a urządzeniem monitoringu (TCP + TLS)

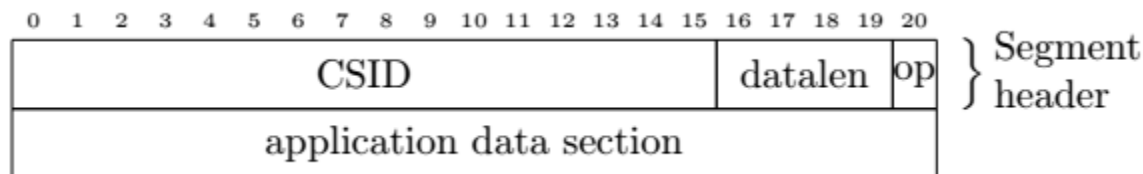
9.2.1 Diagram zadań i kolejek RTOS oraz ich powiązanie z platformą implementacyjną urządzenia monitoringu



9.2.2 Komunikaty przesyłane w ramach komunikacji TCP

- Komunikacja typu duplex

9.2.2.1 Ogólna struktura segmentów TCP przesyłanych między urządzeniem monitoringu a serwerem

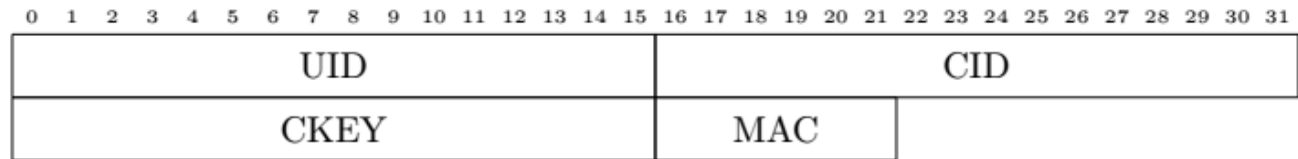


- CSID - camera session identifier - identyfikuje sesję urządzenia monitoringu w systemie działającym na serwerze
- datalen - długość sekcji danych aplikacji
- op - kod określający typ operacji
 - Bit MSB określa czy odbierany segment należy traktować jako żądanie, czy jako odpowiedź

- 0 - żądanie
- 1 - odpowiedź
- application data section - zawiera dane o strukturze zależnej od typu operacji

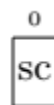
9.2.2.2 Struktury sekcji danych aplikacji dla różnych typów operacji

9.2.2.2.1 Application registration data section (op = APP_CONTROL_OP_REGISTER)



- UID - user identifier
- CID - camera identifier
- CKEY - camera key
- MAC - MAC address (WiFi STA)

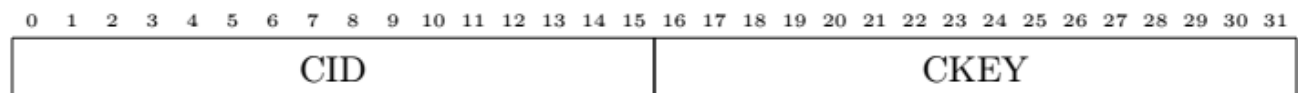
9.2.2.2.2 Application unregistration data section (op = APP_CONTROL_OP_UNREGISTER)



Sekcja ta składa się tylko z jednego bajtu:

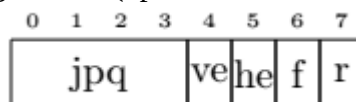
- sc (success) - określa czy operacja się powiodła
 - 0 - nie powiodła się
 - 1 - powiodła się

9.2.2.2.3 Application initcomm data section (op = APP_CONTROL_OP_INITCOMM)



- CID - camera identifier
- CKEY - camera authentication key

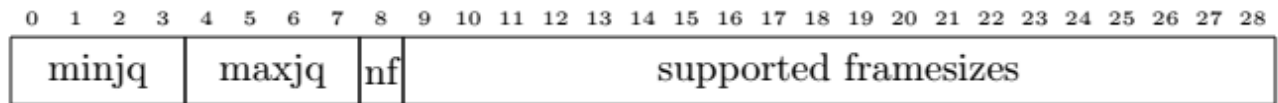
9.2.2.2.4 Application camera config section (op = APP_CONTROL_OP_CAM_GET_CONFIG)



- jpg - jpeg quality (compression parameter) - the lower it is, the higher is the compressed image quality
- ve - vflip enable
- he - hmirror enable

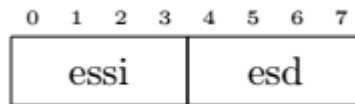
- f - framesize (compatible with espressif__esp32-camera/driver/include/sensor.h:framesize_t defs)
- r - result - one of the following:
 - APP_CONTROL_CAM_CONFIG_RES_OK = 0x0
 - APP_CONTROL_CAM_CONFIG_RES_JPEG_QUALITY_UNSUPPORTED = 0x1
 - APP_CONTROL_CAM_CONFIG_RES_FRAMESIZE_UNSUPPORTED = 0x2

9.2.2.2.5 Application camera caps section (op = APP_CONTROL_OP_CAM_GET_CAPS)



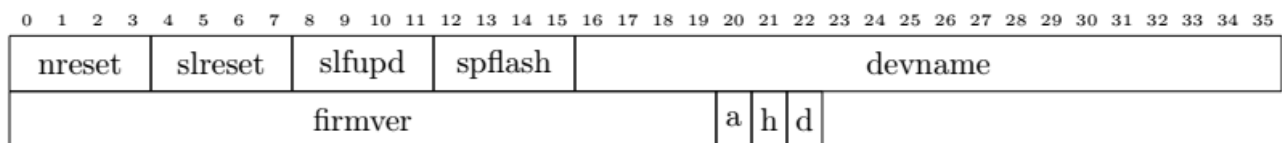
- minjq - minimum supported jpeg quality
- maxjq - maximum supported jpeg quality
- nf - number of supported framesizes
- supported framesizes buffer of uint8_t values, each representing a framesize compliant with the espressif__esp32-camera/driver/include/sensor.h:framesize_t defs

9.2.2.2.6 Application sched section



- essi - energy saving starts in seconds
- esd - energy saving duration seconds

9.2.2.2.7 Application device info section



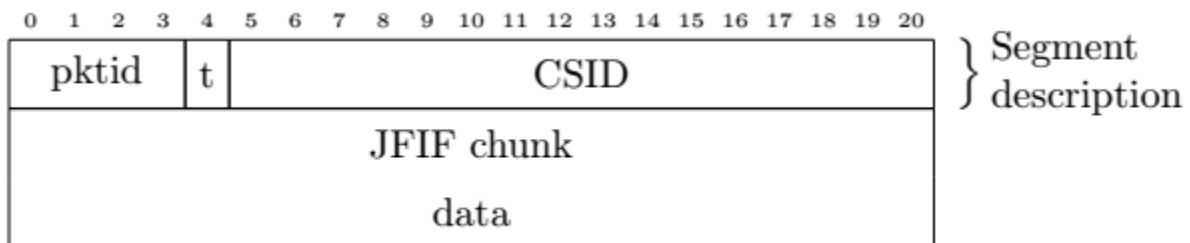
- nreset - number of device resets
- slreset - seconds since last reset
- slfupd - seconds since last firmware update
- spflash - seconds since production flash
- devname - device name
- firmver - firmware version
- a - avc support flag
- h - hevc support flag
- d - dma flag

Nazwa	Kod żądania	Czy ESP32 wspiera?	Czy narzędzie testów esp32 wspiera i testy przechodzą?	Czy serwer wspiera?	Czy aplikacja mobilna obsługuje?
APP_CONTROL_REGISTER	0x01				/
APP_CONTROL_INITCOMM	0x03				
APP_CONTROL_UNREGISTER	0x02				
APP_CONTROL_OP_CAMERA_UNCONDITIONAL_STREAM	0x07				
APP_CONTROL_OP_CAMERA_UNCONDITIONAL_STREAM	0x08				
APP_CONTROL_OP_LOGS_MODE_MINIMAL	0x13				
APP_CONTROL_OP_LOGS_MODE_COMPLETE	0x14				
APP_CONTROL_OP_SOFTWARE_DEVICE_RESET	0x11				
APP_CONTROL_OP_ENERGY_SAVING_SHUTDOWN_ANALYSER	0x0B				
APP_CONTROL_OP_ENERGY_SAVING_W	0x0C				

Nazwa	Kod żądania	Czy ESP32 wspiera?	Czy narzędzie testów esp32 wspiera i testy przechodzą?	Czy serwer wspiera?	Czy aplikacja mobilna obsługuje?
AKEUP_ANALYSER					
APP_CONTROLLER_OP_NOP	0x00				
APP_CONTROLLER_OP_ANALYSE	0x09	in progress			
APP_CONTROLLER_OP_CAMERA_CONFIG	0x05				
APP_CONTROLLER_OP_CAMERA_SETTINGS_CONFIG	0x04				
APP_CONTROLLER_OP_CAMERA_GET_CAPS	0x06				
APP_CONTROLLER_OP_GET_DEVICE_INFO	0x12				
APP_CONTROLLER_OP_CAMERA_STREAM_ANALYSER_STREAM	0x0A				
APP_CONTROLLER_OP_ENERGY_SAVING_SHUTDOWN_NETWORK	0x0D				
APP_CONTROLLER_OP_ENERGY_SAVING_SHUTDOWN_NETWORK_SCHEDULED	0x0E				
APP_CONTROLLER_OP_ENERGY_SAVING_SCHEDULED_SLEEP	0x0F				
APP_CONTROLLER_OP_OTA	0x10				

9.3 Strumieniowanie mediów wideo - między urządzeniem monitoringu a serwerem (UDP + DTLS)

9.3.1 Struktura segmentów UDP wysyłanych przez urządzenie monitoringu do serwera



- pktid - numer segmentu
- t (chunk type) - typ segmentu
 - JFIF_INTERMEDIATE_CHUNK = 0x01
 - JFIF_FIRST_CHUNK = 0x01
 - JFIF_LAST_CHUNK = 0x02
 - JFIF_ONLY_CHUNK = 0x03
- CSID - camera session identifier
- data - fragment bufora danych JFIF pochodzących od sterownika OV2640. Jest stałej długości 16354 bajtów. W przypadku segmentu typu JFIF_LAST_CHUNK lub JFIF_ONLY_CHUNK = 0x03, także jest tej długości, przy czym znaczenie mają jedynie bajty znajdujące się nie dalej niż znacznik EOI (*End of Image*)

###