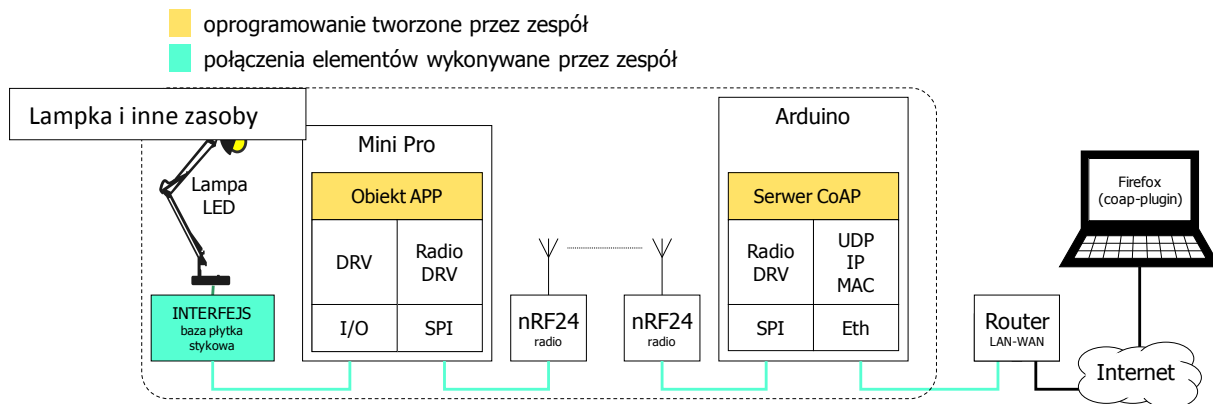


Obiekty Internetu Rzeczy (OBIR)
Lato 2017
Specyfikacja zadania projektowego (Z3)

Specyfikacja ogólna:

Zrealizować system udostępniający opisane niżej zasoby za pomocą protokołu CoAP. Należy przyjąć architekturę jak na rys. 1. W szczególności, wszystkie rozszerzenia sprzętowe powinny być przyłączone do węzła Arduino Mini Pro. Dla wspieranych zasobów należy zaprojektować URI, oraz – tam, gdzie nie jest to oczywiste lub doprecyzowane w niniejszej specyfikacji – ich stan i reprezentację. System ma współpracować z wtyczką Copper (klient CoAP) dla przeglądarki Mozilla, w zakresie wynikającym z podanych niżej możliwości stworzonej przez Zespół implementacji protokołu CoAP. Serwer powinien umieć generować różne kody odpowiedzi, stosownie do sytuacji. W przypadku błędu, serwer powinien zwracać payload diagnostyczny. Do systemu należy dołączyć opisujący go dokument.

Można korzystać ze (a) „standardowych” (dostarczanych z Arduino IDE) bibliotek dla platformy Arduino, oraz (b) bibliotek, których użyto do wykonania zadań na ćwiczeniach laboratoryjnych. Można także korzystać z fragmentów kodu, użytych na ćwiczeniach laboratoryjnych (np. fragmentów zawartych w instrukcjach do ćwiczeń). Poza wymienionymi elementami kodu, całe oprogramowanie niezbędne do realizacji projektu musi być stworzone samodzielnie przez Zespół projektowy. Kod źródłowy powinien być dobrze skomentowany (średnio 30-50% linii kodu powinno być opatrzonych komentarzem).



Rys. 1. Architektura systemu

Zakres wsparcia protokołu CoAP:

1. Obsługa wiadomości NON (GET i/lub PUT, zależnie od potrzeb dla danego zasobu). Obsługa opcji Content-Format, Uri-Path, Accept. Obsługa tokena i MID.
2. Obsługa opcji Observe (dla zasobów oznaczonych poniżej jako „obserwowalne”). Wysyłanie odpowiedzi CON (z retransmisją).

Udostępniane zasoby:

1. Zasób opisujący pozostałe zasoby. Ścieżka /.well-known/core. Ścieżki i atrybuty pozostałych zasobów powinny być określone przez Zespół. GET: pobranie reprezentacji zasobu (w reprezentacji CoRE Link Format), zasób duży (ścieżki i atrybuty należy dobrać tak, aby długość reprezentacji zasobu wynosiła ok. 60B).
2. Lampka. Lampka może być włączona lub wyłączona. GET: pobranie bieżącego stanu włączenia/wyłączenia lampki. PUT: włączanie/wyłączanie.
3. Pokrętło (potencjometr). Stan potencjometru przyjmuje wartości od 0 do 1023, gdzie 0 oznacza wartość minimalną, a 1023 – wartość maksymalną. GET: pobranie bieżącego stanu potencjometru, zasób obserwowalny, zasób o długim czasie dostępu.
4. Zestaw trzech metryk (statystyk) określających jakość połączenia internetowego między Arduino UNO a klientem Copper. Metryki powinny być zaprojektowane przez Zespół. GET: pobranie reprezentacji metryk.

Odbiór projektu:

1. Demonstracja I: interakcja z klientem Copper według uprzednio przygotowanego przez Zespół skryptu (listy). Wszystkie wymienione w tej specyfikacji funkcjonalności powinny zostać zademonstrowane. Wymagane funkcjonalności powinny zostać zebrane w osobnej liście; przy nich powinny znaleźć się odniesienia do odpowiednich pozycji ze skryptu demonstracyjnego.
2. Demonstracja II: interakcja z klientem Copper w sposób określony przez prowadzących.
3. Omówienie architektury oprogramowania: zaprezentowanie przez Zespół krótkiej prezentacji (5 slajdów) + dyskusja z prowadzącymi.
4. Omówienie wybranych przez prowadzących fragmentów kodu źródłowego.
5. Przekazanie prowadzącym płyty CD z wynikami projektu.

Zawartość płyty CD:

1. Dokument opisujący system (architektura oprogramowania: komponenty, interfejsy, przyjęte rozwiązania programistyczne, sposób testowania systemu i wykonane testy, skrypt dla celów demonstracji, zrzuty ekranu pokazujące interakcję z wtyczką Copper).
2. Pełny, skomentowany kod źródłowy.

Kryteria oceny:

1. Zapewnienie wymaganej funkcjonalności.
2. Sprawna i kompletna demonstracja, wg dobrze przygotowanego skryptu.
3. Znajomość kodu źródłowego.
4. Jakość dokumentu.