Mikołaj Baran, 241128 Jakub Aniszewski, 241133

prowadzący: Dominik Żelazny

#### Laboratorium Urządzeń Peryferyjnych

#### Ćwiczenie 14 - Bluetooth

### 1 Cel ćwiczenia

- 1. Wykryć adaptery BT podłączone do PC.
- 2. Użyć wybranego adaptera do zdalnego wyszukiwania urządzeń BT.
- 3. Pobrać adres MAC wybranego (wyszukanego w pkt. 2) urządzenia.
- 4. Dokonać autoryzacji obu urządzeń:
  - po stronie urządzenia BT autoryzować PC
  - po stronie PC autoryzować urządzenie BT.
- 5. Uruchomić urządznie BT w tryb pracy transferu plików.
- 6. Przesłać plik tekstowy do urządzenia BT.
- 7. Przesłać plik graficzny do urządzenia BT.

## 2 Wstęp

Bluetooth jest standardem bezprzewodowej komunikacji o krótkim zasięgu pomiędzy różnymi urządzeniami elektronicznymi, najczęściej sa to telefony komórkowe. Jest to otwarty standard opisany w **IEEE 802.15.1** a jego specyfikacja techniczna obejmuje trzy klasy mocy nadawczej:

- Klasa 1 100 mW ma zasięg do 100 m
- Klasa 2 2.5 mW ma zasieg do 10 m
- Klasa 3 1 mW ma zasięg do 1 m

Najczęściej spotykana jest klasa druga. Standard ten wykorzystuje fale radiowe w nielicencjonowanym paśmie **ISM 2,4GHz**, które początkowo przeznaczone było dla zastosowań przemysłowych, naukowych oraz medycznych, natomiast obecnie z części tych zakresów korzystają systemy bezprzewodowej transmisji danych.

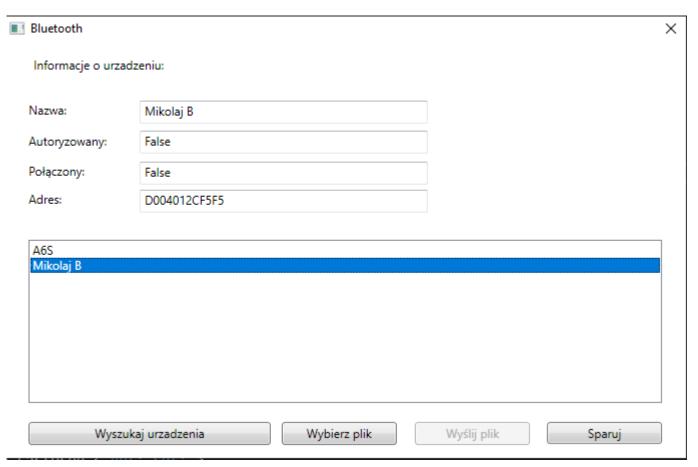
W pierwszej wersji tego standardu maksymalna przepustowość wynosiła 21 kb/s. W obecnej wersji, tj. Bluetooth 5.0 Transfer wynosi do 50 Mb/s. Standard Bluetooth ma własny model dla protokołów, składający się z 3 warstw. Najważniejszą i najniższą z nich jest warstwa radiowa, odpowiedzialna za transport danych od urządzenia typu master do slave oraz w drugą stronę. Kolejną jest warstwa Baseband Layer która pakuje luźne bity w ramki. Ostatnia warstwa L2CAP przyjmuje pakiety od wyższych warstw i dzieli je na ramki, zajmuje się multipleksją oraz demultipleksją złożonych pakietów, a także zajmuje się wymaganiami na jakość usługi.

OBEX – Object Exchange to protokół komunikacyjny, określający procedury wymiany danych binarnych między urządzeniami. Rozwojem i utrzymaniem specyfikacji zajmuje się Infrared Data Association. Protokół jest używany do przesyłania danych w takich technologiach jak: Bluetooth, USB czy RS232. Specyfikacja protokołu OBEX opiera się na architekturze klient – serwer. Klient wykorzystuje zaufane medium transportowe do połączenia się z serwerem w celu zażądania transmisji obiektów. Przesyłane obiekty są zapisane w formacie binarnym.

### 3 Opis programu

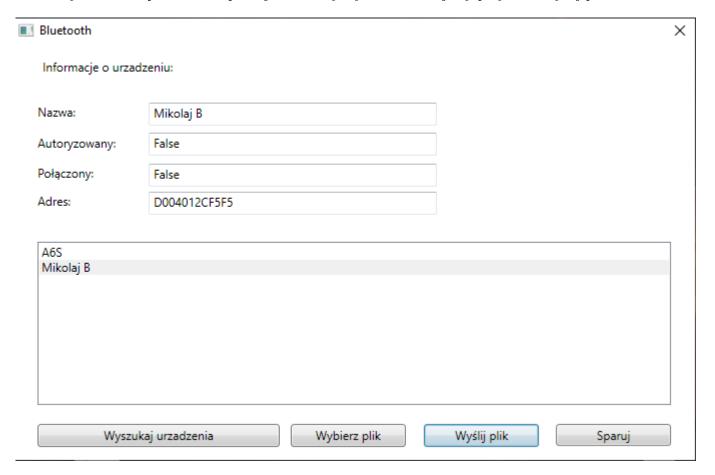
Program był pisany w języku C# w środowisku Visual Studio przy użyciu biblioteki 32feet.NET oraz transmisji plików OBEX.

Na początku należy użyć opcji "Wyszukaj urządzenia" aby program znalazł inne urządenia z włączoną usługą Bluetooth w pobliżu.



Kiedy już je znajdzie możemy wybrać urządzenie które nas insteresuje i zobaczyć bardziej szczegółowe informacje, czy został zautoryzowany, czy jest połączony oraz adres MAC urządzenia.

Po wybraniu urządzenia oraz pliku jaki chcemy wysłać odblokujemy przycisk "Wyślij plik".



Teraz po jego wciśnięciu na innym urządzeniu otrzymujemy pytanie czy chcemy odebrać dany plik.

## **♯**‡ Bluetooth ∨

# Plik przychodzący

test.txt



Funkcja działa zarówno dla plików tekstowych jak i obrazów.

## 4 Najważniejsze funkcje w programie

• Funkcja wyszukująca urządzenia.

```
void bg_DoWork(object sender, DoWorkEventArgs e)
{
    List < Device > devices = new List < Device > ();
    InTheHand.Net.Sockets.BluetoothClient bc = new InTheHand.Net.Sockets.BluetoothClient()
    InTheHand.Net.Sockets.BluetoothDeviceInfo[] array = bc.DiscoverDevices();
    int count = array.Length;
    for (int i = 0; i < count; i++)
    {
         Device device = new Device(array[i]);
         devices.Add(device);
    }
    e.Result = devices;
}</pre>
```

Funkcja na początku tworzy listę urządzeń. Device to napisana na potrzeby programu klasa która przechowuje informacje o urządzeniu. Następnie tworzymy instancję klasy BluetoothClient ponieważ posiada ona metodę DiscoverDevices która wyszukuje urządzenia w pobliżu. Znalezione urządzenia zapisujemy w tablicy. Potem tylko iterujemy przez tę tablice i używając konstruktora klasy Device przechowujemy w programie informacje o odnalezionym urządzniu w bardziej przestępny sposób i dodajemy utworzone obiekty do listy.

• Funkcja do wysyłania plików.

Na początku towrzymy obiekt klasy *Uri* któremu w konstruktorze przekazujemy adres urządzenia oraz ścieżkę do naszego pliku. URI (Uniform Resource Identifier) czyli ujednolicony identyfikator zasobów jest standardem internetowym umożliwiającym łatwą identyfikację zasobów w sieci. Następnie towrzymy instancję klasy *ObexWebRequest* aby móc użyć jej metody *ReadFile* która wczytuje zawartość naszego pliku do otwartego strumienia. Potem aby uzyskać tworzymy *ObexWebResponse* na który rzutujemy wynik metody *GetResponse*, zamykamy strumień i zwracamy kod informujący o powodzeniu lub niepowodzeniu wysyłania.

• Funkcja pozwalająca wybierać pliki z systemu.

```
private void browseBtnClick(object sender, RoutedEventArgs e)
{
    Microsoft.Win32.OpenFileDialog dlg = new Microsoft.Win32.OpenFileDialog();
    dlg.FileName = "File To Send";
    dlg.DefaultExt = "*.*";

    Nullable<bool> result = dlg.ShowDialog();

    if (result == true)
    {
        FilePath = dlg.FileName;
        sendBtn.IsEnabled = true;
    }
    else
        sendBtn.IsEnabled = false;
}
```

Tworzymy instancję klasy *OpenFileDialog* która przedstawia wspólne okno dialogowe, umożliwiające użytkownikowi określenie nazwy pliku, który ma zostać otwarty. Ustawiamy domyślną nazwe pliku oraz jego rozszerzenie na dowolne. Teraz wyświetlamy to okno przy użyciu *ShowDialog* i zapisujemy czy użytkownik klikną w nim **OK** czy **Cancel**. na podstawie tego zapamiętujemy ścieżkę do otwartego pliku i aktywujemy lub dezaktywujemy przycisk "Wyślij plik".

#### 5 Wnioski

Podczas zajęć laboratoryjnych mieliśmy okazję zapoznać się z działaniem technologii Bluetooth z perspektywy programisty. Dowiedzieliśmy się, że do przesyłania obiektów przez Bluetooth można wykorzystać protokół OBEX. Przy tworzeniu aplikacji wykorzystaliśmy również bibliotekę 32feet.NET.

Napisany przez nas program pozwala na realizację wszystkich podpunktów wymienionych w celach ćwiczenia oprócz autoryzacji urządzenia. Autoryzacja nie funkcjonowała poprawnie, ponieważ metoda *pairBtnClick*, która odpowiada za parowanie urządzeń powinna zostać rozszerzona o możliwość podania przez użytkownika kodu PIN potrzebnego do sparowania urządzeń.