

1. PRZEGLĄD LITERATURY

1.1. Wstęp

W dzisiejszych czasach, gdy technologia w zakresie elektroniki i informatyki jest bardzo rozwinięta, zaczęły pojawiać się rozwiązania ułatwiające rehabilitację osób w sposób dokładny, mierzalny oraz niejednokrotnie ciekawy i zachęcający do terapii. Tematem tego projektu jest realizacja odpowiedniej komunikacji pomiędzy urządzeniem służącym do rehabilitacji (w tym przypadku interfejs człowiek-komputer) a komputerem PC. Wykorzystywaną w tym celu technologią na całym świecie jest protokół Bluetooth oraz sieć Wi-Fi. W tym rozdziale przedstawione zostaną informacje, które pomogą w realizacji tego projektu. W podpunkcie 1.2 ukazana zostanie w sposób ogólny technologia Bluetooth, jak powstała, w jakim celu ją stworzono oraz kiedy można ją wykorzystywać. W podpunkcie 1.3 wyjaśniono, w jaki sposób przebiega komunikacja w protokole Bluetooth oraz jak zbudowana jest sieć oparta na tym protokole. W kolejnym podpunkcie zawarto informacje na temat modułów Bluetooth w różnych standardach i przedstawiono budowę i cechy przykładowego modułu. W podpunkcie 1.5. zawarto informację na temat dostępnych bibliotek w języku Java, których można użyć do stworzenia aplikacji desktopowej. W ostatnim podpunkcie tego rozdziału przedstawiono, w jaki sposób można przeprowadzić komunikację aplikacji z serwerem TCP/IP.

1.2. Technologia Bluetooth

Technologia Bluetooth to globalny standard bezprzewodowej komunikacji między urządzeniami opisany w specyfikacji IEEE 802.15.1. Został utworzony w 1994 roku przez firmę Ericsson Mobile Communications jako bezprzewodowa alternatywa dla przesyłania danych poprzez ich wymianę za pomocą transmisji radiowych. Celem było opracowanie taniego łącza radiowego o niskiej mocy, umożliwiającego połączenie telefonów komórkowych z akcesoriami. Głównym przedmiotem zainteresowania było stworzenie bezprzewodowego zestawu słuchawkowego [1].

Nazwa pochodzi od duńskiego króla Haralda Sinozębego (w języku angielskim Harold Bluetooth), który pomógł zjednoczyć się walczącym frakcjom, obecnie były to obszary Norwegii, Szwecji i Danii. Podobnie technologia Bluetooth została stworzona, aby umożliwić łączność i współpracę różnych produktów i branży.

Ze względu na to, iż połączenia kablowe są niewygodne, a łącza na podczerwień wymagają bezpośredniej widoczności pomiędzy nadajnikiem a odbiornikiem rozważono użycie łącza radiowego, które pozwala na wyeliminowanie kabli i nie wymaga bezpośredniej widoczności. Cechami, które miały charakteryzować technologię Bluetooth są niski pobór prądu, niewielki zasięg oraz mały koszt. Dzięki temu kreatorzy umożliwili całkowite wyeliminowanie przesyłania danych za pomocą kabli [2].

Technologia Bluetooth pozwala na połączenie nawet siedmiu różnych urządzeń, których wysyłane sygnały nie zakłócałyby się wzajemnie. Dzieje się tak ze względu na to, iż sygnały przesyłane są skokowo – w standardzie Bluetooth 1.0 przeskakują 1600 razy na sekundę pomiędzy poszczególnymi kanałami częstotliwości, natomiast średnia wartość z uwzględnieniem wszystkich standardów wynosi około 800. W

praktyce zaleca się jednak, aby liczba tych urządzeń w obrębie jednej sieci nie przekraczała 4, co gwarantuje najlepszą jakość przesyłania.

Bluetooth wykorzystuje pasmo częstotliwości od 2400 do 2483,5 MHz. Zakres ten podzielony jest na 79 kanałów rozmieszczonych co 1 MHz (od 2402 do 2480 MHz), co zapewnia maksymalną prędkość transferu 721 kb/s. W przypadku najnowszego standardu Bluetooth 4.0 występuje 40 kanałów o przepustowości 2 MHz, co gwarantuje maksymalny transfer danych do 3 Mb/s. Dane przenoszone w postaci pakietów i transmituje każdy pakiet na jeden z 79 wyznaczonych kanałów [3].

W przypadku standardu Bluetooth występuje zjawisko kompatybilności wstecznej. Co prawda połączenie dwóch urządzeń o różnych standardach powoduje wolniejszy transfer danych, jednakże jest to spore udogodnienie, ponieważ zgodność protokołów gwarantuje, że przekazane dane dotrą w takiej samej formie, w jakiej zostały wysłane.

Urządzenia wyposażone w technologię Bluetooth mogą działać w trybie komunikacji dwukierunkowej half-duplex lub full-duplex. Ten tryb połączenia umożliwia jednoczesne przysyłanie różnych danych pomiędzy urządzeniami bez obniżenia prędkości transferu. Przykładowym urządzeniem pracującym w trybie full-duplex jest telefon komórkowy – podczas rozmowy telefonicznej, w przypadku rozmówców, przesyłane są jednocześnie dźwięk do słuchawki oraz dźwięk z mikrofonu. Tryb half-duplex umożliwia przesyłanie danych, czyli ich odbiór oraz wysyłkę na przemian, czego wynikiem jest spadek transferu pakietów.

Jak każda forma bezprzewodowego przesyłania danych, również Bluetooth jest narażony na różne formy ataku. Istnieją pewne rodzaje zabezpieczeń zależnych od rodzaju urządzenia. Najbardziej popularnym rozwiązaniem jest utworzenie listy zaufanych urządzeń, które mogą przysyłać pakiety danych bez zapytania o zgodę. Jeśli urządzenie spoza listy będzie próbowało nawiązać połączenie, przesłanie jakichkolwiek informacji nie odbędzie się bez wyrażenia na to zgody użytkownika. Co więcej, urządzenia często oferują funkcję niewidoczny, dzięki której urządzenia nie będzie widoczne w sieci. Dzięki temu, że w większości przypadków użytkownicy dodatkowo proszeni są o wyrażenie zgody na zapisania, bądź zainstalowanie pliku, ryzyko zainfekowania telefonu, czy smartfonu maleje [4].

1.3. Komunikacja w protokole Bluetooth

W technologii Bluetooth można wyróżnić węzły typu master oraz typu slave. Węzeł typu master i siedem węzłów typu slave tworzy pikosieć, która pracuje w oparciu o system TDM (Time Division Multiplexing). W pikosieci, poza siedmioma węzłami typu slave, może istnieć do 255 węzłów, które pracują w synchronizacji z węzłem master. Urządzenia te mogą jedynie otrzymywać sygnał aktywacyjny lub nawigacyjny od węzła typu master, nie biorą one udziału w wymianie danych. Pikosieci mogą być ze sobą połączone za pomocą węzła bridge. Połączone ze sobą pikosieci to scatternet.

Wszystkie urządzenia w pikosieci są zsynchronizowane z zegarem urządzenia, którym jest master. Możliwość wymiany danych istnieje tylko między węzłem master i slave. Węzeł typu master kontroluje zegar i określa, które urządzenie i w którym slotcie czasowym może się z nim komunikować. Węzeł master może równocześnie nawiązać połączenie z większą liczbą węzłów slave, natomiast każdy z węzłów typu slave może

komunikować się tylko z jednym węzłem master. Dzięki opisanemu wyżej podziałowi węzłów, koszty technologii uległy zmniejszeniu [3].

Węzeł typu master jako jedyny może zainicjować komunikację Bluetooth. Węzeł slave odbiera nadany pakiet oraz przesyła informację zwrotną z potwierdzeniem odbioru. Węzeł typu master nie ma obowiązku na to odpowiadać. Jeśli obydwa węzły nie mają więcej danych do przesłania, proces komunikacji kończy się. Wcześniej, w dowolnym momencie, na przykład jeśli odebrany pakiet zawiera więcej, niż dopuszczalna norma błędów, istnieje możliwość przerwania komunikacji [5].

Stos protokołu Bluetooth składa się z:

- Grupy protokołów transportowych – dotyczą asynchronicznych transmisji wymiany danych oraz okresowych transmisji komunikacji głosowej. Pozwalają na wzajemną lokalizację urządzeń oraz na przesyłanie danych z jednego urządzenia do drugiego, za pomocą interfejsu radiowego Bluetooth. Do grupy transportowej należą protokoły radiowe, pasma podstawowego, menedżera połączenia, połączenia logicznego i adaptacji oraz interfejsu kontrolera hosta;
- W pierwszej kolejności dane przekazywane są przez warstwę protokołu kontroli połączenia logicznego i adaptacji (L2CAP), który umożliwia multipleksację protokołów, podział pakietów oraz utrzymuje odpowiedni poziom usług. Następną warstwą jest warstwa menedżera połączenia (LMP), której zadaniem jest ustalenie parametrów łącza radiowego między urządzeniami Bluetooth oraz sterowanie mocą. Warstwa pasma podstawowego definiuje interfejs radiowy Bluetooth, przydziela urządzeniom węzeł master, bądź slave oraz określa reguły współdzielenia interfejsu radiowego przez kilka urządzeń;
- Grupy protokołów pośredniczących – umożliwiają poprawną pracę aplikacji opartych na połączeniach Bluetooth. W skład tej grupy wchodzi protokoły związane z Internetem, np. PPP, protokoły aplikacji bezprzewodowych, przesyłania obiektów adaptowane ze standardu IrDA;
- Grupy aplikacji – w jej skład wchodzi aplikacje, korzystające z połączeń Bluetooth, np. aplikacja zestawienia połączenia modemowego [6].

1.4. Moduły Bluetooth

Na rynku dostępnych jest wiele modułów Bluetooth dla mikrokontrolerów. Można je podzielić ze względu na:

- Interfejs:
 - UART (USART),
 - SPI,
 - I²C,
 - I²S.

Najpopularniejszym i najprostszym interfejsem jest UART (Universal Asynchronous serial Receiver and Transmitter). Jest to interfejs szeregowy wykorzystujący do komunikacji dwie linie sygnałowe: nadawczą i odbiorczą.

- Zgodność ze standardem Bluetooth:
 - Bluetooth v. 1.1 (1.2),

- Bluetooth v. 2.0 (2.1),
- Bluetooth v. 3.0,
- Bluetooth v. 4.0.

Główną różnicą pomiędzy specyfikacją Bluetooth 2.0 i 3.0 jest zwiększenie transferu danych za pomocą wykorzystania fizycznej warstwy standardu 802.11. W kolejnej wersji - 4.0 duży nacisk położono na ograniczenie zapotrzebowania na energię [7].

Różnice pomiędzy poszczególnymi standardami standardu Bluetooth przedstawiona została w tabeli 1.4.1.

Tab. 1.4.1. Porównanie standardów Bluetooth [8]

Standard	v. 1.1	v. 1.2	v. 2.0	v. 2.1 plus EDR (Enhanced Data Rate)	v. 3.0	v. 4.0
Przesyłanie głosu	+	+	+	+	+	+
Wyciszenie rozmowy	+	+	+	+	+	+
Ponowne wybieranie ostatniego numeru	+	+	+	+	+	+
Szybkie prędkości transmisji	-	-	+	+	+	+
Niższe zużycie energii	-	-	+	+	+	+
Ulepszone parowanie (bez kodu PIN)	-	-	-	+	+	+
Większe bezpieczeństwo	-	+	+	+	+	+
Bluetooth Low Energy	-	-	-	-	-	+
Wsparcie NFC	-	-	+	+	+	+

Najbardziej popularnym standardem wśród modułów jest Bluetooth 2.1. Moduły są stosunkowo tanie, zaczynają się już od ok. 25zł (HC-06). W projekcie ważnym aspektem jest niski pobór energii przez moduł, przez co lepszym wydaje się więc moduł zgodny z Bluetooth 4.0 LE.

Opis modułu Bluetooth opracowany został na podstawie modułu Adafruit Bluefruit LE UART Friend. Jak sama nazwa wskazuje moduł obsługuje technologię Bluetooth Low Energy oraz wyposażony jest w interfejs komunikacyjny UART. Moduł wyróżnia się wysoką jakością wykonania oraz dobrą dokumentacją producenta.

Specyfikacja modułu:

- układ sterowany mikrokontrolerem wyposażonym w rdzeń ARM Cortex M0,
- częstotliwość pracy 16 MHz,
- pamięć Flash: 256 kB,
- pamięć SRAM: 32 kB,
- komunikacja UART:
 - prędkość: 9600 bps,
 - kontrola przepływu Flow Control (CTS + RTS),
- wbudowany stabilizator napięcia 3,3 V,
- wspiera komendy AT,

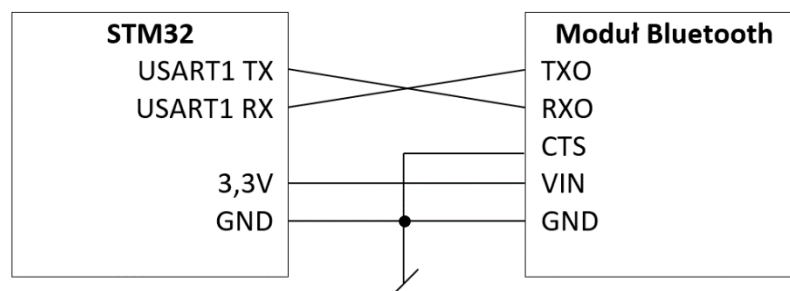
- wymiary płytki: 32x21x5 mm,
- masa około 3,5 g,
- cena około 85 zł.

Moduł wyposażony jest w następujące wyprowadzenia:

- VIN – pin do zasilania 3,3-16 V,
- GND – masa,
- TXO – pin komunikacyjny wyjściowy interfejsu UART. Wartość dla wysokiego poziomu logicznego to 3,3 V;
- RXI – pin komunikacyjny wejściowy interfejsu UART. Dla wysokiego poziomu logicznego konieczne jest przyłożenie napięcia 3-5 V;
- CTS – pin wykorzystywany do sprzętowej kontroli przepływu, służy do informowania modułu, że może wysłać dane do mikrokontrolera. Ponieważ by nie będziemy potrzebowali wyżej wymienionej funkcji, pin zostanie połączony z masą;
- RTS – pin wykorzystywany do kontroli przepływu. Pin będzie miał niski poziom napięcia, gdy będzie można wysłać dane do modułu. W naszym przypadku wyprowadzenie nie będzie wykorzystywane;
- MOD – pin wykorzystywany do wyboru trybu pracy modułu pomiędzy wysyłaniem danych (DATA MODE, niski poziom logiczny) a konfiguracją urządzenia (COMMAND MODE, wysoki poziom logiczny). Tryb pracy można zmienić również za pomocą fizycznego przełącznika na module. W trybie konfiguracji wysyła się polecenia za pomocą komend Hayes/AT. W trybie tym można zmieniać np. moc czy nazwę pod jaką widziany moduł będzie dla innych urządzeń.

Pozostałe wyprowadzenia służą do aktualizacji firmware-u, resetu modułu itp. Opisane one są w dokumentacji producenta [9].

Schemat połączenia modułu Bluetooth wyposażonego w interfejs komunikacyjny UART z mikrokontrolerem STM32 przedstawiony został na rysunku 1.4.1.



Rys. 1.4.1. Połączenie modułu Bluetooth z mikrokontrolerem STM32

1.5. Biblioteki Java implementujące protokół Bluetooth

Najczęściej wykorzystywaną biblioteką w języku Java realizującą komunikację Bluetooth jest BlueCove. Pierwotnie została ona opracowana przez dział badań w firmie Intel, obecnie prace nad nią prowadzone są przez wolontariuszy.

BlueCove jest implementacją specyfikacji JSR-82 (Java Specification Request) znanej pod nazwą Java APIs for Bluetooth Wireless Technology (JABWT) [10].

Specyfikacja JSR-82 zdefiniowana została przez Java Community Process i odpowiedzialna jest za zapewnienie standardu do tworzenia aplikacji Bluetooth w języku Java. Dodatkowo jest to specyfikacja otwarta i niezastrzeżona. JSR-82 ukrywa złożoność stosu protokołu Bluetooth, poprzez eksponowanie prostego zbioru API w języku Java.

Aby stworzyć aplikację implementującą standard Bluetooth w Javie, wykorzystując standard JSR-82, należy spełnić dwa warunki wstępne. Pierwszym warunkiem jest posiadanie stosu Bluetooth zgodnego z JSR-82. Potrzebna jest implementacja JSR-82, aby z powodzeniem testować i rozwijać aplikacje Bluetooth wykonane w Javie. Dodatkowo w tym warunku potrzebne jest symulowane środowisko takie jak Sun Java Wireless Toolkit lub prawdziwy stos Bluetooth w Javie taki jak ElectricBlue albo Avelink. Symulator pozwala na testowanie aplikacji w symulowanym środowisku, bez dostępu do prawdziwego modułu Bluetooth. Drugi warunek obowiązuje tylko wtedy, gdy nie używa się symulatora. Aby przetestować aplikację na realnym urządzeniu ponad stosem Bluetooth, potrzebny jest włączony system Bluetooth. Może to być Bluetooth USB Dongle podłączony do systemu, na którym działa aplikacja [11].

Biblioteka BlueCove działa na każdej wirtualnej maszynie Javy (JVM – Java Virtual Machine) zaczynając od wersji 1.1 po nowsze [10]. Dla usprawnienia działania tej biblioteki dodawane są dwa moduły. Pierwszy to BlueCove-GPL, który jest dodatkowym, licencjonowanym modulem dla wsparcia działania biblioteki na Linux BlueZ. Drugi to BlueCove JSR-82 Emulator, który jest dodatkowym modulem dla BlueCove, aby symulować działanie stosu Bluetooth [12].

Dla poprawnego działania biblioteki BlueCove trzeba spełniać kilka wymogów. Po pierwsze jest to oprogramowanie WIDCOMM (Broadcom) wersja 1.4.2.10 SP5 albo wyższa. Jest to oprogramowanie potrzebne przy komunikacji Bluetooth. Drugim jest program BlueSoleil wersja 1.6.0, 2.3 albo 3.2.2.8. BlueSoleil jest to narzędzie do zarządzania zawartością telefonu za pośrednictwem Bluetooth. Dodatkowo potrzebny jest stos Bluetooth dla Microsoft, który znajduje się w systemie operacyjnym Windows XP SP2 lub nowszym i Windows Mobile 2003 lub nowszym. Kolejnym wymogiem, obowiązującym jeśli pracujemy na systemie Mac OS X 10.4, jest posiadanie mikroprocesora PowerPC lub Intel – podstawowy (Bluetooth wersja 1.2) lub późny (od wersji 2.0.2). Jeśli pracujemy na systemie Linux potrzebny jest stos Bluetooth BlueZ. Jednym z ostatnich wymogów jest posiadanie Javy w wersji 1.1 lub nowszej do realizacji binarnej i Javy 1.4 lub nowszej do kompilowania. Ostatnim jest posiadanie dwóch modułów Bluetooth. Pierwsze obsługiwane przez stos Bluetooth dla WIDCOMM, BlueSoleil lub Microsoft, drugie potrzebne do komunikacji [10].

Przykładem projektu, w którym wykorzystana została opisana biblioteka BlueCove, jest iSMS. Jest to system, którego działanie polega na otrzymywaniu powiadomień o wysłaniu/odebraniu wiadomości SMS, a także informacji o połączeniach korzystając z komputera, który za pomocą protokołu Bluetooth łączy się z urządzeniem mobilnym posiadającym system Android [13].

1.6. Komunikacja TCP/IP

W celu umożliwienia wymiany informacji pomiędzy komputerem a serwerem niezbędne jest wykorzystanie warstwowej struktury protokołów komunikacyjnych opartej o model TCP/IP. Ma on szerokie zastosowanie m.in. w systemach rehabilitacyjnych. Przykładowo w [14] autorzy stworzyli urządzenie zawierające szereg modułów pozwalających na lokalną lub zdalną rehabilitację dłoni w oparciu o ćwiczenia sprawdzające chwytanie, ściskanie, rozszerzanie i rotację dłoni oraz stukanie palcami w odpowiednie pola. Powyższy system działa w oparciu o architekturę klient – serwer wykonaną przy użyciu języka C++, natomiast jako bazę danych wykorzystano SQLite. Po włączeniu urządzenia następuje połączenie z serwerem TCP/IP, który automatycznie przechodzi w tryb nasłuchu. W chwili rozpoczęcia terapii użytkownik wykonuje serię ćwiczeń, których wyniki otrzymuje się za pomocą licznych sensorów zawartych w urządzeniu. Następnie uzyskane rezultaty wysyłane są na serwer i odbierane przez terapeutę w postaci pliku w formacie .csv w momencie uruchomienia aplikacji klienckiej. Pobrane dane zawierają wyniki ostatnich sesji rehabilitacji pacjentów i są wizualizowane w postaci wykresów. Klient może zbudować żądania do tworzenia zapytań w celu otrzymania interesujących go danych, jednakże w celu ochrony integralności danych nie jest możliwe usunięcie żadnych danych z serwera TCP/IP z aplikacji terapeuty.

Ze względu na niewielką ilość przesyłanych danych pomiędzy aplikacją a serwerem nie jest konieczne ciągle utrzymywanie połączenia. W takiej sytuacji wykorzystuje się protokół HTTP (Hypertext Transfer Protocol), który realizuje tzw. żądania (requests) poprzez metody GET oraz POST. W języku JAVA do wysyłania takich żądań służy klasa `URLConnection` [15], w której przesyłanie danych następuje na podstawie adresu URL oraz `payload'a` (paczki) z danymi. Każda instancja tej klasy służy do wykonywania pojedynczego żądania, ale zasadnicze połączenie z serwerem HTTP może być przezroczysto dzielone pomiędzy inne instancje. Dodatkowo do przesyłania danych można użyć formatu wymiany danych JSON (JavaScript Object Notation), który jest lekkim formatem tekstowym, bazującym na języku JavaScript. Istotną zaletą takiego podejścia jest nastawienie na zasoby (obiekty) identyfikowane przez URL, bardzo czytelne wiadomości o małym rozmiarze, łatwo interpretowalne przez klienta HTTP. Wykonany w taki sposób system można nazwać systemem RESTful od stylu architektury REST (Representational State Transfer) [16].

Protokół HTTP po otrzymaniu żądania od aplikacji klienckiej zwraca informację w postaci numeru ze sposobem realizacji tegoż zapytania. Numer ten znajduje się tuż za wersją protokołu HTTP i przed słownym opisem kodu odpowiedzi. Wyróżnia się następujące kategorie kodów:

- 1xx – informacyjne,
- 2xx – powodzenia,
- 3xx – przekierowania,
- 4xx – błędu aplikacji klienta,
- 5xx – błędu serwera HTTP [17].

Tego typu rozwiązania są szeroko wykorzystywane w sieciach następnej generacji (Next Generation Networks) opartych na przesyłaniu pakietów do użytkowników

końcowych niezależnie od sieci dostępowej, urządzenia i technologii dostępu. Autorzy w [18] wskazują, iż w nowoczesnych systemach coraz częściej wykorzystuje się architekturę REST, a w celu obsługi żądań protokół HTTP, ponieważ tego typu technologie webowe są bardzo proste, nie wymagają skomplikowanych bibliotek, mogą w prosty sposób zostać ponownie użyte i sprawdzają się szczególnie w systemach rozproszonych. Ponadto HTTP jest głównym i podstawowym protokołem transferowym obsługującym żądania za pomocą metod m.in. GET i POST, które zostały wymienione wcześniej oraz PUT, DELETE, przez co jest powszechnie wykorzystywany w wielu systemach internetowych.

1.7. Technologie w rehabilitacji

Rehabilitacja jest formą kompleksowej pomocy, która jest skierowana do osób niepełnosprawnych fizycznie lub psychicznie. Celem rehabilitacji jest powrót do pełnej, bądź częściowej sprawności fizycznej lub psychicznej. Zajęcia rehabilitacyjne mogą przebiegać w szpitalu lub w domu, prowadzone przez rehabilitanta.

Wyróżnić można wiele sposobów rehabilitacji osób niepełnosprawnych, zalicza się do nich:

- rehabilitację leczniczą – wykorzystuje biologiczne metody oddziaływania na osobę z dysfunkcją organizmu,
- rehabilitację społeczną – przywraca możliwość normalnego sposobu życia w naturalnym środowisku,
- rehabilitację zawodową - ułatwia osobie niepełnosprawnej zdobycie pracy, dzięki umożliwieniu jej korzystania z poradnictwa zawodowego, szkolenia zawodowego oraz pośrednictwa pracy,
- rehabilitację pedagogiczną – proces nauczania dostosowany do możliwości i ograniczeń osób niepełnosprawnych [19].

Dzięki rehabilitacji, osoby niepełnosprawne dążą do uzyskania godziwego oraz lepszego życia, w poczuciu bezpieczeństwa społecznego, jak również zawodowego. Dlatego, aby uczynić rehabilitację jak najbardziej wydajną oraz efektywną, stosuje się nowe technologie, mające na celu wspomaganie rozwoju rehabilitacji. Wiele firm inwestuje w produkcję sprzętów oraz medykamentów, kupuje nowoczesne linie technologiczne lub opracowuje własne, innowacyjne projekty.

Do przykładów nowatorskich produktów rehabilitacyjnych można zaliczyć cyberoko, stworzone przez inżynierów Politechniki Gdańskiej, czy też stoły do rehabilitacji, wanny do masażu, fotele rehabilitacyjne, itp. Zaczęto również zastanawiać się nad takimi rozwiązaniami jak interaktywne kostki, dzięki którym można mieć możliwość ilościowej oceny postępów osoby niepełnosprawnej, która jest poddana rehabilitacji [20].

WYKAZ LITERATURY

- [1] A. Daniluk, „Bluetooth. Praktyczne programowanie”, Wydawnictwo Uniwersytetu Marii Curie – Skłodowskiej w Lublinie, 2012.
- [2] „Witryna technologii Bluetooth”, 2016. [Online]. Dostępne na: <https://www.bluetooth.com/what-is-bluetooth-technology/bluetooth>. [Odwiedzono: 28.03.2016].
- [3] „Witryna Tutorial-Reports”, 2013. [Online]. Dostępne na: <http://www.tutorial-reports.com/wireless/bluetooth/tutorial.php>. [Odwiedzono: 28.03.2016].
- [4] N. Rymsza, „Technologia Bluetooth – czym jest, jak powstała i dlaczego tak się nazywa?”, 2014. [Online]. Dostępne na: <http://smartdeal.pl/wszystko-o-technologie-bluetooth/>. [Odwiedzono: 28.03.2016].
- [5] M. Jaworska, „Bluetooth gotowy na IoT”, 2015. [Online]. Dostępne na: <http://elektronikab2b.pl/technika/27015-bluetooth-gotowy-na-iot>. [Odwiedzono: 29.03.2016].
- [6] B. A. Miller i C. Bisdikian, „Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications (2nd Edition)”, *Prentice Hall PTR*, s. 65-71, 2001.
- [7] „Witryna speed guide”, 2016. [Online]. Dostępne na: <http://www.speedguide.net/faq/what-is-the-difference-between-bluetooth-20-30-460>. [Odwiedzono: 28.03.2016].
- [8] V. Zhang, „Comparison of Bluetooth Different Versions”, 2012. [Online]. Dostępne na: <http://wistao.blogspot.de/2012/09/comparison-of-bluetooth-different.html>. [Odwiedzono: 28.03.2016].
- [9] „Witryna firmy Adafruit”, 2015. [Online]. Dostępne na: <https://learn.adafruit.com/introducing-the-adafruit-bluefruit-le-uart-friend/introduction>. [Odwiedzono: 28.03.2016].
- [10] „Witryna Google Code projektu BlueCove”, 2016. [Online]. Dostępne na: <https://code.google.com/archive/p/bluecove/wikis/Documentation.wiki>. [Odwiedzono: 26.03.2016].
- [11] „Witryna specyfikacji JSR-82”, 2006. [Online]. Dostępne na: <http://www.jsr82.com/jsr-82-basics/>. [Odwiedzono: 28.03.2016].
- [12] „Witryna projektu BlueCove”, 2008. [Online]. Dostępne na: <http://bluecove.org/>. [Odwiedzono: 26.03.2016].
- [13] N. S. Gunay, „iSMS: Computer texting for personal and business use”, 2015 *International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, s. 1-4, 2015.
- [14] D. Pani, G. Barabino, A. Dessi, I. Tradori, M. Piga, A. Mathieu i L. Raffo, „A Device for Local or Remote Monitoring of Hand Rehabilitation Sessions for

Rheumatic Patients”, 2014 *IEEE Journal of Translational Engineering in Health and Medicine*, 2014.

- [15] „Dokumentacja Java, Klasa HttpURLConnection”, 2016. [Online]. Dostępne na: <http://docs.oracle.com/javase/7/docs/api/java/net/HttpURLConnection.html>. [Odwiedzono: 26.03.2016].
- [16] T. Fredrich, „What Is REST?”. [Online]. Dostępne na: <http://www.restapitutorial.com/lessons/whatisrest.html>. [Odwiedzono: 26.03.2016].
- [17] „Witryna organizacji IETF”, 2016. [Online]. Dostępne na: <http://www.ietf.org/assignments/http-status-codes/http-status-codes.xml>. [Odwiedzono: 26.03.2016].
- [18] F. Belqasmi, Ch. Fu i R. Glitho, „RESTful Web Services for Service Provisioning in Next Generation Networks: A Survey”, *IEEE Communications Magazine* (Volume: 49 , Issue: 12), s. 66-73, 2011.
- [19] „Witryna portalu ABC Zdrowie”, 2016. [Online]. Dostępne na: <https://portal.abczdrowie.pl/rehabilitacja>. [Odwiedzono: 23.05.2016].
- [20] E. Wesołowska, „Miliardy w rehabilitacji. Polskie firmy inwestują w nowe technologie”, 2011. [Online]. Dostępne na: <http://biznes.gazetaprawna.pl/artykuly/552668,miliardy-w-rehabilitacji-polskie-firmy-inwestuja-w-nowe-technologie.html>. [Odwiedzono: 23.05.2016].