AKADEMIA MORSKA

SZCZECIN

Wydział Mechaniczny

Instytut ……………..

Numer ewidencyjny ……………….

Data pobrania tematu pracy …………….

Data zdania pracy ………………

PRACA DYPLOMOWA

INŻYNIERSKA

|  |  |  |  |
| --- | --- | --- | --- |
| Dyplomant | Marek Jakóbiak | | Nr albumu: 24135 |
| Kierunek/Specjalność | Mechanika i Budowa Maszyn/Eksploatacja Siłowni Okrętowych | | |
| Promotor | dr inż. Marcin Mąka | Ocena: | |
| Recenzent |  | Ocena: | |
| Egzamin dyplomowy - data |  | | |

TEMAT: Opracowanie aplikacji do odbioru informacji z radiostacji MF/HF

Dyplomant………………… Promotor………………… Dziekan…………………

**AKADEMIA MORSKA W SZCZECINIE**



**WYDZIAŁ NAWIGACYJNY**



Marek Jakóbiak

**PRACA DYPLOMOWA INŻYNIERSKA**

**Temat: Opracowanie aplikacji do odbioru informacji z radiostacji MF/HF**

Praca wykonana w Instytucie ………………………….

pod kierunkiem …………………………….

Szczecin 2019

Spis treści

[Wykaz użytych skrótów i symboli 6](#_Toc4245182)

[Wstęp 8](#_Toc4245183)

[1.Przysałanie informacji w eterze 11](#_Toc4245184)

[2.System DSC 12](#_Toc4245185)

[4.Projekt aplikacji do odbioru informacji z radiostacji MF/HF 13](#_Toc4245186)

[4.1 Zastosowane technologie i biblioteki 13](#_Toc4245187)

[4.1.1. WPF 13](#_Toc4245188)

[4.1.2 C# 14](#_Toc4245189)

[4.1.3 WPF 4.5 14](#_Toc4245190)

[Bibliografia 15](#_Toc4245191)

# Wykaz użytych skrótów i symboli

|  |  |  |
| --- | --- | --- |
| Skrót | Określenie | |
|  | W języku angielskim | W języku polskim |
| COMSAR | Sub-Committee on Radiocommunciations and Search and Rescue |  |
| IMO | International Maritime Organization | Międzynarodowa Organizacja Morska |
| DSC |  |  |
| HF |  |  |
| MF |  |  |
| VHF |  |  |
| GMDSS | Global Maritime Distress and Safety System |  |
| SSB | Single SideBand | Modulacja jednowstęgowa |
| MMSI |  |  |
| MVVM |  |  |
| API | Application Programming Interface | Interfejs programowania aplikacji |
| XAML | Extensible Application Markup Language |  |
| XML | eXtensible Markup Language |  |
| HTML | HyperText Markup Language |  |
| UI | User Interface |  |
| UX | User eXperience |  |
| GDI+ | Graphics Device Interface+ |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Wstęp

Wykorzystanie fali elektromagnetycznej do transmisji danych, wytworzonej po raz pierwszy w 1886 roku przez Heinricha Hertza, nieodwracalnie zmieniło obraz współczesnej cywilizacji. Chociaż odkrycie samego radia w roku 1895 zawdzięczamy Nikoli Tesli, to jednak dopiero włoski wynalazca Gugellelmo Marconi zdołał wzbudzić powszechne zainteresowanie nowym sposobem na przesyłanie informacji nadając 27 lipca 1896 roku sygnał z dachu Poczty Głównej w Londynie do oddalonego o kilometr odbiornika. Prezentacja wywołała wśród widzów ogromne zainteresowanie co zaowocowało, w 1899 roku, pierwszą udaną próbną transmisją sygnału przez kanał La Manche, a w 1901 emisją, z Anglii do Kanady, tym razem konkretnej już informacji: poprzez Ocean Atlantycki nadano literę „S”. Wybuch I Wojny Światowej w roku 1914 przyspieszył nieco rozwój badań nad radiokomunikacją, jednakże jego apogeum przypada dopiero na lata dwudzieste XX wieku. Na przestrzeni lat 1922-1933 na terenach wszystkich państw europejskich oraz w całych Stanach Zjednoczonych powstały liczne radiostacje nadające regularne programy i audycje, natomiast przystępna cena radioodbiorników sprawiła, że urządzenie stało się powszechne nawet w domach średniozamożnych rodzin. Stworzyło to niespotykaną wcześniej możliwość dotarcia do mas, co w krótkim czasie zostało z powodzeniem wykorzystane w celach politycznych: komunikacja radiowa stała się narzędziem propagandowym.

Równolegle do zastosowań cywilnych, radiofonia została użyta również w celach militarnych. Wraz z jej rozprzestrzenieniem się zarówno na lądzie jak i na morzu, zaczęła uwydatniać się potrzeba zabezpieczania transmitowanych informacji. Obawa przed dostaniem się w niepowołane ręce wrażliwych danych zapoczątkowała metodę kodowania nadawanych komunikatów oraz badania nad szyfrowaniem. Niemniej jednak, dopiero wprowadzenie cyfrowych nadajników operujących na danych w formie binarnej przyczynił się do rozwoju kryptologii i nadał jej szczególnego znaczenia.

Wraz z końcem Zimnej Wojny zaczęto skupiać się na rozwoju technologii mającej na celu poprawę bezpieczeństwa na wodach morskich i śródlądowych oraz zmniejszenie szkód wyrządzanych przez błędy nawigacyjne. Opracowując nowy system przyjęto założenie, że podstawą sprawnego i bezkolizyjnego przemieszczania się wszelkiego rodzaju jednostek pływających jest zachowanie efektywnej komunikacji zarówno pomiędzy tymi jednostkami, jak i pomiędzy poszczególnymi punktami nadzorującymi ich manewry. Jednym z kroków milowych w zakresie bezpieczeństwa ruchu okrętów było stworzenie – w roku 1992 – Światowego Morskiego Systemu Łączności Alarmowej i Bezpieczeństwa (Global Maritime Distress and Safety System – GMDSS). Siedmioletni okres jego wdrażania zakończył się z dniem 01.02.1999, o czym oficjalnie poinformowano podczas IV Sesji Podkomitetu d.s. Radiokomunikacji, oraz Poszukiwań i Ratownictwa (Sub-Committee on Radiocommunciations and Search and Rescue – COMSAR), i Międzynarodowej Organizacji Morskiej (The International Maritime Organization – IMO), która odbyła się w lipcu tego samego roku.

Tak długi okres implementacji umożliwił przeprowadzenie wielu miarodajnych doświadczeń z zakresu różnych dziedzin, które obejmowały, między innymi, eksploatację systemu, weryfikację procedur operacyjnych a nawet konstrukcję specjalistycznych urządzeń GMDSS. Jednymi z najważniejszych spośród zrealizowanych prób były te, które dotyczyły analiz pracy systemu w warunkach rzeczywistych, a w szczególności funkcjonowania podsystemów umożliwiających prowadzenie łączności oraz alarmowanie w sytuacjach zagrożenia.

Jednym z wielu systemów zawierających się w GMDSS, służącym do komunikacji w sytuacjach alarmowych, jest cyfrowe wywołanie selektywne (Digital selective calling – DSC). Jest to przystawka wchodząca w skład radiostacji MF/HF, znanej również pod nazwą radia modulacji jednowstęgowej (Single SideBand - SSB) ze względu na sposób w jaki nadawane są dane. Warto również zaznaczyć, że radiostacja umożliwia transmisję Tx i Rx, czyli można za jej pomocą nadawać oraz odbierać dane i komunikację dźwiękową. Przed pojawieniem się przystawki DSC, jedną z największych wad był brak możliwości adresacji korespondencji do danego radioodbiornika. Było to wyjątkowo kłopotliwe na akwenach o wysokiej gęstości ruchu, w tym w portach. Często w takich rejonach ilość dostępnych kanałów nie była wystarczająca. Sytuacja uległa zmianie w momencie Wykorzystanie tego typu radiostacji, które znajdują zastosowanie przede wszystkim w sytuacjach wymagających zautomatyzowanego nawiązywania łączności w radiokomunikacji morskiej, jest odpowiednie dla DSC przewidzianego do pracy w paśmie pośredniofalowym (MF; 2Mhz), krótkofalowym (HF; 4Mhz, 6Mhz, 8Mhz, 12Mhz, 16Mhz) oraz ultrakrótkofalowym (VHF; 156 – 174 Mhz). Jego głównym zadaniem jest przesyłanie wywołań alarmowych w sytuacjach niebezpieczeństwa (distress), niemniej jednak odpowiedzialny jest on również za automatyczne nawiązywanie łączności w celach publicznych oraz eksploatacyjnych. Stało się to możliwe dzięki zaimplementowaniu w ramkę komunikatu pola kategorii. Szeroki wybór kategorii pozwala radiooperatorom w prosty sposób określić powód nadania informacji oraz jej charakter. W celu zminimalizowania ryzyka zaistnienia problemów z odbiorem i przetwarzaniem komunikatów, format ramki przesyłanego komunikatu zawartego w dokumentacji technicznej jest ściśle ustalony normami Międzynarodowej Organizacji Morskiej. Ramka komunikatu posiada również pole służące do kontroli poprawności przesyłanych informacji – znak detekcji błędu (wielkość 8 bitów). Mechanizm ów opiera się na kontroli parzystości otrzymanych danych, dzięki której w wypadku wykrycia niepoprawności, radiostacja wysyła zapytanie o powtórzenie komunikatu w celu dokonania ewentualnej korekty błędnie otrzymanej informacji.

# 1.Przysałanie informacji w eterze

# 2.System DSC

# 4.Projekt aplikacji do odbioru informacji z radiostacji MF/HF

## 4.1 Zastosowane technologie i biblioteki

W dobie szeroko rozpowszechnionego systemu operacyjnego Windows (74,44% na dzień 16.03.2019r [1]) na desktopach, stworzenie aplikacji w oparciu o rozwiązaniu firmy Microsoft wydaje się być rozsądnym pomysłem. Dedykowanym temu systemowi operacyjnemu platforma programistyczna to .NET Framework, w której skład wchodzą kompilatory języków wysokiego poziomu C++/CLI, C#, Visual Basic .NET i J#. Wybrałem drugi z nich ze względu na wnikliwe zaznajomienie się z nim w czasie toku studiów. Pozostało jedynie określić wzorzec projektowy i bibliotekę. W tego typu projektach obecnie dominuje wzorzec MVVM (Model-View-ViewModel). Ograniczyło to mój wybór biblioteki do WPF (Windows Presentation Foundation) i UWP (Universal Windows Platform). Dylemat został rozstrzygnięty poprzez kompatybilność obu rozwiązań. Drugie rozwiązanie można uruchomić jedynie na systemie operacyjnym Windows 10, oraz konsoli Xbox One/Xbox One X, a pierwsze na systemach od Windows XP wzwyż. W następnych podrozdziałach przybliżę nieco technologie i zewnętrzną bibliotekę użyte w implementacji mojego rozwiązania.

### 4.1.1. WPF

Wraz z rozwojem sprzętu graficznego, a tym samym oczekiwań konsumentów co wyglądu aplikacji, w Microsofcie została podjęta decyzja, aby opracować nowy produkt, który pozwoli uwolnić się od ograniczeń GDI+ (podsystem Windows XP; API wystawione poprzez zestaw klas C++ służący do wyświetlania informacji) oraz Windows USER (komponent systemów operacyjnych Microsoft Windows służący do tworzenia prostych interfejsów użytkownika), jednocześnie zachowując wysoki poziom produktywności znany użytkownikom Windows Forms. Wynikiem była biblioteka Windows Presentation Foundation (WPF) mająca łączyć najlepsze funkcje takich systemów jak Windows Forms (wydajność programisty), HTML (deklaratywne znaczniki), Adobe Flash (zaawansowane wsparcie dla animacji), oraz DirectX (3D i akceleracja sprzętowa)[3].

### XAML

XAML (eXtensible Application Markup Language) to względnie prosty, deklaratywny język znaczników zainspirowany językami XML (eXtensible Markup Language) i HTML (HyperText Markup Language). Jest mechanizmem do wykorzystywania API frameworka .NET poprzez oddzielenie części odpowiedzialnej za interakcję z użytkownikiem od części logicznej programu[2]. Pozwala to na niezależną od siebie pracę UI/UX (User Interface/User eXperience) Designerów oraz C# developerów. Potrzeba takiej rozdzielności wywodzi się ze skuteczności metodyk tworzenia aplikacji internetowych (np. SCRUM) i chęci ich wdrożenia również do aplikacji desktopowych. Część widoczna dla użytkownika, tzw. frontend, komunikuje się z częścią logiczną programu głównie za pomocą dwóch mechanizmów. Pierwszym z nich jest tzw. code behind, czyli plik .cs doczepiony do pliku .xaml, w którym można pisać kod w języku C#. Dzięki temu można pisać logikę aplikacji w widoku i odsłaniać zmienne przed innymi klasami w tej samej przestrzeni nazw za pomocą modyfikatorów dostępu. Jest to niezalecany sposób pisania aplikacji, niezgodny ze wzorcem architektonicznym MVVM, który w code behind dopuszcza jedynie walidację wprowadzanych danych. Decydowanie się na pisanie logiki aplikacji w pliku .cs doczepionym do .xml jest uzasadnione jedynie w przypadku naprawdę małych projektów, gdzie model MVVM byłby przerostem formy nad treścią. Drugim, preferowanym mechanizmem, jest binding (wiązanie). Nazwę swoją zawdzięcza nazwie funkcji, przez którą jest realizowany. Otóż w XAML, w dowolnej właściwości kontrolki, można użyć instrukcji „Binding [nazwa]” (np. ComboBox SelectedValue="{Binding StopBitsValue}"), aby móc w danej przestrzeni nazw odnieść się do uprzednio zdefiniowanego zasobu. Następnie, po ustawieniu nowej wartości w kodzie C#, należy poinformować interfejs użytkownika o zmianie poprzez wykonanie metody PropertyChanged z interfejsu INotifyPropertyChanged, aby został odświeżony. Jest to niestety jeden z niewielu mankamentów rozwiązania Microsoftu i jego zrozumienie jest kluczowe sprawnego posługiwania się WPFem[3].

## C#

Jest to język zaprojektowany w latach 1998-2001 przez firmę Microsoft, będący bogatą implementacją paradygmatu obiektowego. Pozwala deweloperom na stworzenie szeregu aplikacji uruchamianych na .NET Framework. Charakteryzuje się elegancką, typowaną statycznie, składnią pozwalającą na uniknięcie błędów wynikłych ze złego określenia typów. Dzięki temu kompilator jest w stanie sprawdzić, czy nie próbujemy użyć ciągu znaków (string) w funkcji przystosowanej do liczb (integer).

# Bibliografia

1. <http://gs.statcounter.com/os-market-share/desktop/worldwide>, dostęp 16.03.2019r.
2. <https://docs.microsoft.com/en-gb/dotnet/framework/wpf/advanced/xaml-overview-wpf>, dostęp 18.03.2019r.
3. Nathan A., WPF 4.5 Księga Eksperta, Helion, Gliwice 2015
4. <https://docs.microsoft.com/pl-pl/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework>, dostęp 21.03.2019r.