

***Piotr Szpila***

***092100***

**Informatyka, studia stacjonarne I stopnia, semestr 4**

Projekt inżynierski I pt.:

**Projekt i implementacji aplikacji do generowania i edycji**

**ontologii OWL Lotniska.**

**Rzeszów, 2017 r.**

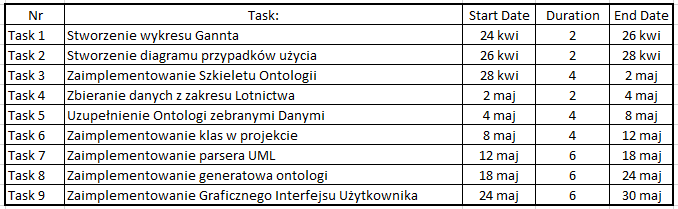
**1. CEL I ZAKRES PROJEKTU**

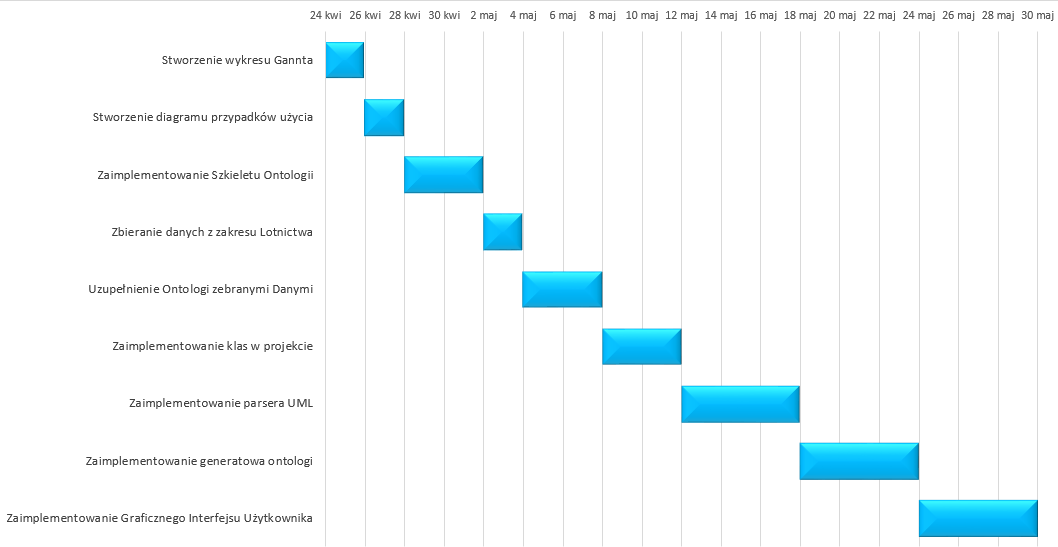
Projekt został stworzony w celu modelowania lotów lotniska międzynarodowego oraz samolotami w posiadaniu spółki. Program przechowuje dane na temat lotnisk pomiędzy którymi regularnie kursują loty spółki. Projekt zawiera kompletną dokumentację przebiegu budowy ontologii oraz rozwoju aplikacji na przestrzeni 01.03.2017 do 23.05.2017.

Zakres projekt obejmuje:

* Zaprojektowanie ontologii OWL
* Stworzenie aplikacji desktopowej pozwalającej na podstawowe operacje na ontologii takie jak: wczytanie danych, dodawanie, usuwanie, edycję danych z ontologii.
* Wykonanie potrzebnych do dokumentacji diagramów, schematów grafów
* Stworzenie dokumentacji

Wykres Gantta przedstawia proces powstawania projektu, poszczególne etapy zostały podzielone na zadania, następujące po sobie w uporządkowany sposób, tak aby tworzyły spójną strukturę





**2. OPIS OBSZARU TEMATYKI**

Tematyką projektu jest modelowanie lotów przez kontrolera lotów   
w przykładowej spółce lotniczej. Obszar ten został przeze mnie wybrany,   
z powodu rozwijającego się lotniska w pobliżu Rzeszowa oraz firm i zakładów przy nim. Zagadnienie lotniska ma również duży potencjał do rozwijania   
o kolejne klasy oraz systemy rządzące tym skomplikowanym mechanizmem.

Projekt przedstawia zarządzanie lotami krajowymi oraz między narodowymi. Przeloty pomiędzy lotniskami krajowymi oraz zagranicznymi wykorzystują samoloty posiadane przez spółkę. Na potrzeby uproszczenia modelowanej dziedziny samoloty zostały podzielone na dwa typu wąskokadłubowe wykonujące przeloty pomiędzy lotniskami krajowymi oraz samoloty szerokokadłubowe wykonujące przeloty zagraniczne.

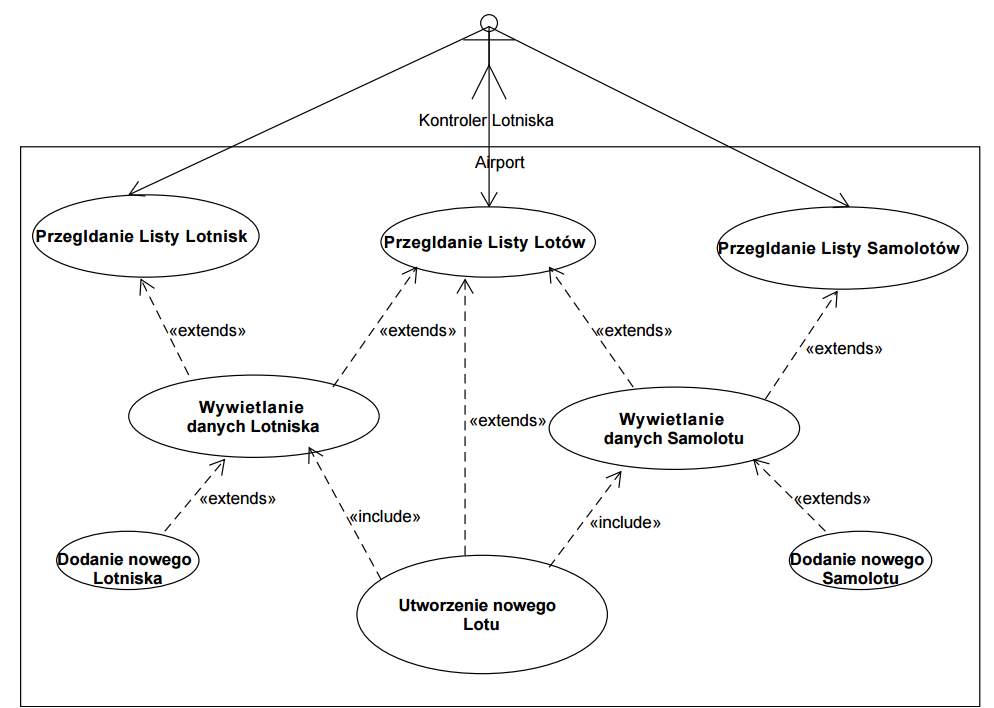
Modelowana dziedzina została odwzorowana w ontologii OWL stworzonej   
w programie Protege. Program ten został stworzony do budowania prostych jak   
i złożonych aplikacji bazujących na ontologii OWL oraz rozbudowanych inteligentnych systemów. Wybrałem właśnie ten program ten z kilku głównych powodów:

* Jest programem darmowym oraz open-source, aktywnie rozwijanym przez środowiska akademickie oraz korporacyjne.
* Wspiera najnowszy język OWL 2 ze standardem W3C Konsorcjum.
* Program jest bazowany na języku Java oraz posiada wiele rozszerzeń.

Na projekt składają się:

* Ontologia w języku OWL
* Aplikacja desktopowa wykonana w języku Java 8
* Dokumentacja projektowa wraz z:
  + - Wykresem Gantta
    - Grafem klas ontologii
    - Diagramem klas
    - Schematem przypadków Użycia

Możliwości programu oraz jego zastosowania przedstawia diagram przypadków użycia. Który opisuje działania jakie Kontroler Lotów może wykonać za pomocą aplikacji oraz zapisać w ontologii.



**3. Projekt Ontologii OWL**

Ontologia zawiera w sobie trzy główne klasy:

* „Plane” – dzieli się na podklasy:
  + „Narrow Body”
  + „Wide Body”,

Obie te klasy przechowują indywidua samolotów pasażerskich. Które zawierają dane:

* + - Ilości siedzeń pasażerskich
    - Ilości silników
    - Roku produkcji
    - Unikalnego numeru samolotu (Id)
* „Airport” – dzieli się na podklasy:
  + „Domestic Airport”
  + „International Airport”.

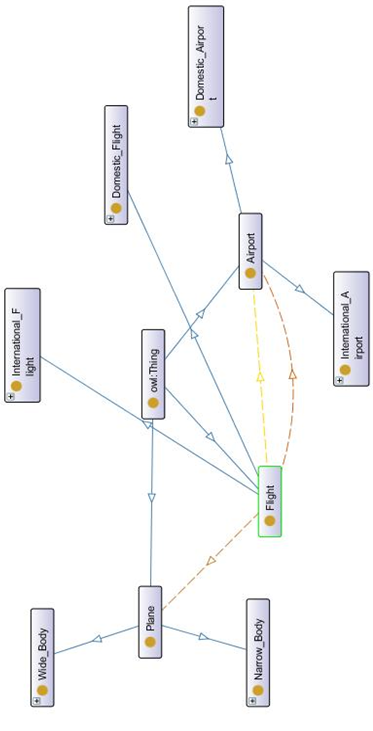
Indywidua tych podklas to lotniska krajowe oraz międzynarodowe oraz zawierają dane o:

* + - Powierzchni lotniska w Hektarach
    - Unikalnym numerze lotniska (Id)
* Flight – dzieli się na podklasy:
  + „Domestic Flight” – indywidua tej podklasy wykorzystują samoloty typu „Narrow Body” oraz kursują pomiędzy lotniskami krajowymi „Domestic Flights”.
  + „Internacional Flight” - indywidua tej podklasy wykorzystują samoloty typu „Wide Body” oraz kursują pomiędzy lotniskami między narodowymi „Internacional Flights”.

Indywidua te zawierają dane o:

* Miejscu Startu typu „Airport”
* Miejscu Docelowym typu „Airport”
* Użytym Samolocie typu „Plane”
* Unikalnym numerze lotu (Id)

Schemat ontologii obrazuje graf:



**4. Wykorzystane metody, narzędzia i technologie**

Projekt został wykonany przy użyciu narzędzie takich jak:

* Protege – Program został w projekcie wykorzystany do stworzenia ontologii oraz wyeksportowania go do pliku typu OWL/XML. Program ten został wybrany przeze mnie, ponieważ jest darmowy oraz często używany w środowisku uniwersyteckim na całym świecie.
* OntoGraph – narzędzie rozszerzające Protege pozwalające na graficzne przedstawienie ontologie w grafie bądź diagramie. Wybrałem to narzędzie, ponieważ jest ono bardzo przyjazne użytkownikowi i łatwe w obsłudze.
* OWLApi – parser plików. owl stworzony przez uniwersytet Stanforda do przetwarzania i obróbki ontologii w języku programowania Java. Wybrałem go ze względu na bogatą dokumentację oraz dedykowane zastosowanie dla tego typu projektów.
* Java 8 –język programowania które najlepiej znam oraz w którym został stworzony program Protege. W projekcie zastosowałem najnowsze rozwiązania takie jak wyrażenia lambda oraz strumienie.
* Maven – narzędzie automatyzujące budowę projektów na platformę Java. Wybrałem je ze względu na precyzyjną kontrolę nad celami budowy projektu oraz łatwość zarządzania bibliotekami potrzebnymi do parsowania ontologii w tym przypadku OWLApi.
* JavaFX z technologią FXML to narzędzie które wykorzystałem do stworzenia Graficznego Interfejsu Użytkownika, w celu przedstawienia danych zawartych w ontologii. Narzędzie to jest mi bardzo dobrze znane, posiada również bogatą dokumentację oraz jest darmowe.
* Loombok – narzędzie usprawniające tworzenie modeli, pozwala na generowanie akcesoriów i muratorów do modelu za pomocą adnotacji.

**5. Opis implementacji aplikacji**

Aplikacja składa się z trzech głównych modułów:

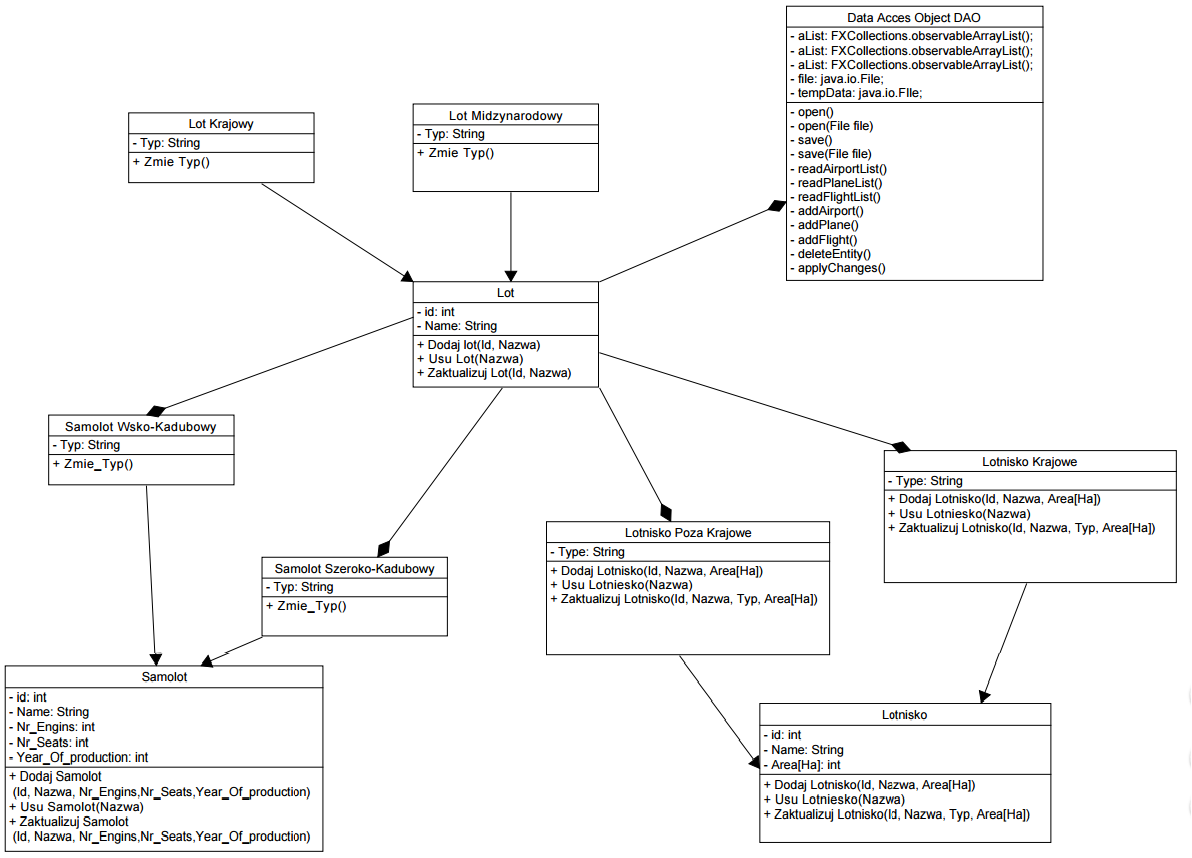
* Pierwszym z nich są trzy modele od wzorujące te zastosowane w ontologii, czyli
  + - Airport,
    - Plane,
    - Flight

Modele zawierają pola do przechowywania informacji o Data Property  
i Object Pro perty z odpowiadających im indywiduów.

* Kolejnym modułem jest klasa DAO (Data Access Object). Klasa ta otwiera połączenie z ontologią polegające na inicjalizację
  + - menagera,
    - rezonera,
    - factory

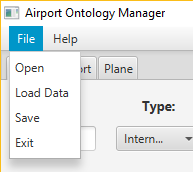
potrzebnych do operacji na ontologie takich jak wczytanie danych   
z ontologii, operacje dodawania indywiduów oraz ich usuwanie. Klasa ta również odpowiada za przeładowanie danych po operacji dodawania lub usuwania indywiduów bądź Data Property lub Object Property.

* Trzecim modułem, z którego składa się projekt to plik FXML definiujący wygląd okna oraz jego kontroler. Ustala on wielkości poszczególnych elementów okna, ich układ oraz identyfikatory używane przez kontroler okna. Kontroler definiuje zachowanie po naciśnięciu na elementy okna,   
  w tym wypadku jest to wywoływanie odpowiednich metod z klasy DAO, wraz z parametrami.



7. Podręcznik użytkownika

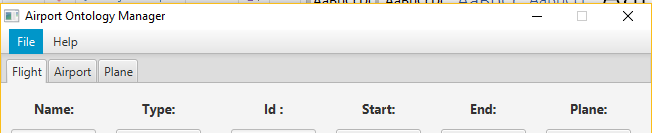
Pasek górny menu



zawira akcje takie jak:

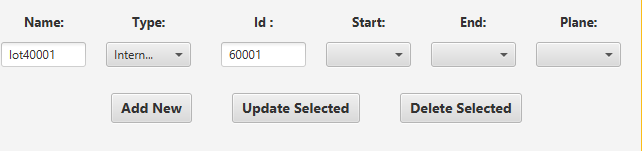
* + Otworzenie pliku z ontologią o rozszerzeniu \*.owl
  + Wczytanie danych z pliku \*.txt
  + Zapisanie ontologi
  + Wyjście z programu

Aplikacja do każdej klasy udostępnia osobną zakładkę:

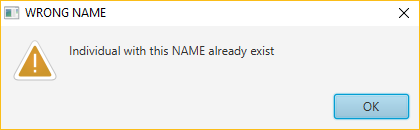


Aplikacja umożliwia takie działania na ontologii jak:

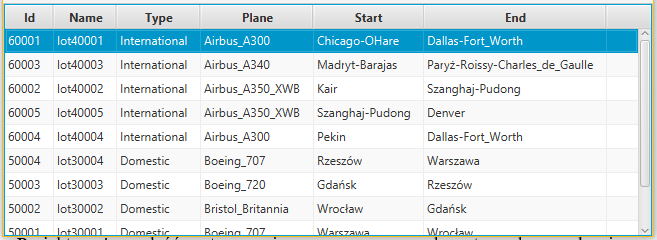
* Wyświetlanie danych
* Dodawanie inwitów oraz właściwości
* Modyfikacja danych
* Usuwanie danych



Program zabezpieczony przed dodawaniem indywiduów   
o powtarzających się nazwach bądź identyfikatora.



Okno tabeli:



Wyświetla dane z wybranej z zakładki, kliknięcie na wybrany wiersz   
w tabeli uzupełnia dane w oknie działania, ułatwiając akcje modyfikacji bądź usuwania.

7. Podsumowanie

Projekt może znaleźć zastosowanie przy uproszczonych systemach zarządzania lotniskiem. Pozwala on na kontrolę z pozycji kontrolera lotów na archiwizację odbytych lotów w obrębie jednej spólki. Kontroller ma do dyspozycji narzędzia takie jak:

* Przeglądanie listy lotów, samolotów, lotnisk.
* Dodawanie lotów, samolotów, lotnisk.
* Edycję lotów, samolotów, lotnisk.
* Usówanie lotów, samolotów, lotnisk.

Działania te obejmują założenia cztery działania CRUD wykorzystywane   
w aplikacjach korzystających z pamięci trwałej. W tym przypadku dane zapisywane są w ontologi OWL która spełnia rolę bazy danych.