

Seminarium Dyplomowe Semestr 7

Zajęcia nr 3

Identyfikacja dziedziny problemu

Mgr inż. Jerzy Stankiewicz

ZAJĘCIA NR 1 - ROZLICZENIE

- **Przygotować harmonogram prac** z wykorzystaniem MS Project (za okres październik 2016 : 31-03-2017r)
- **Utworzyć dokument pracy dyplomowej** (nazwisko imię v1.docx) ze stroną tytułową, proponowanymi rozdziałami (z wygenerowanym spisem treści)
- **Literatura** (na końcu dokumentu) – przedstawić propozycje literatury z dziedzin:
 - Projektowania systemów informatycznych
 - Modelowania systemów
 - Projektowania baz danych
 - Języków programowania
 - Dziedziny tematycznej pracy dyplomowej (normy prawne, dzienniki ustaw itp.)
 - Strony internetowe (ćwiczenia, opisy, przegląd produktów rynkowych o podobnej tematyce itp.)
- **Opracować rozdział wstępny** w zakresie: temat pracy, cel i zakres pracy, wprowadzenie do problemu (ogólne)

- ***Opracować część analityczną pracy dyplomowej w zakresie: szczegółowy opis problemu***

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Wymagania *(Wikipedia)*

- **Wymaganie** – pojedyncza, udokumentowana potrzeba określonego produktu czy usługi albo sposobu ich działania
- Formalnie jest to wykorzystywane powszechniej w inżynierii systemów lub w inżynierii oprogramowania
- Jest to stwierdzenie identyfikujące potrzebne cechy, możliwości, charakterystyki lub jakość systemu, aby był on wartościowy i pożyteczny dla użytkownika
- W klasycznej inżynierii, zbiór wymagań jest wykorzystywany w fazie projektowania nowego produktu. Wymagania pokazują, jakie elementy i funkcje są niezbędne w konkretnym projekcie
- Faza opracowywania wymagań może być poprzedzona studium wykonalności lub koncepcyjną fazą projektu

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Wymagania

- Faza wymagań może być podzielona na:
 - Gromadzenie wymagań (zbieranie wymagań od interesariuszy)
 - Analizowanie (sprawdzenie spójności i kompletności)
 - specyfikowanie (dokumentowanie wymagań)
 - oraz zatwierdzanie (upewnienie się, że wymagania są poprawne)
- Wyniki -> **Raport**: dokument, w którym zbiera się i opisuje wymagania dotyczące przyszłego systemu (powinien być podstawą ustaleń (kontraktu) między klientem a producentem:
 - **Wprowadzenie** – cele, zakres i kontekst systemu – wyniki fazy strategicznej
 - Opis **ewolucji systemu** – opis przewidywanych zmian wymagań wobec systemu
 - Opis **wymagań funkcjonalnych**
 - Opis **wymagań pozafunkcjonalnych**
 - **Model systemu** (zwykle w formie graficznej)
 - **Słownik pojęć** – definicje i opis terminów niezrozumiałych dla jednej ze stron (terminy dotyczące dziedziny i informatyczne), a także terminy niejednoznaczne, zależne od kontekstu.

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Wymagania produktowe a procesowe (Wikipedia)

- Projekty są podmiotem trzech rodzajów wymagań:
 - Wymagania **biznesowe** opisują (w terminologii handlowej) co ma być dostarczone lub wykonane w celu uzyskania wartości
 - Wymagania **produktowe** opisują system lub produkt, który na jeden z możliwych sposobów wypełnia wymagania biznesowe
 - Wymagania **procesowe** opisują procesy, które organizacja musi zrealizować i ograniczenia, które muszą być respektowane
- Wymagania produktowe i procesowe są ściśle powiązane
- Wymagania procesowe często są narzucone jako droga osiągnięcia niektórych wymagań produktowych
- Przykłady:
 - maksymalny wymagany koszt wytworzenia (wymaganie procesowe) może być wymuszone przez maksymalną cenę zbytu (wymaganie produktowe)
 - wymaganie na produkt ma być utrzymane (wymaganie produktowe), ale może być realizowane różnymi metodami, jak np. programowanie obiektowe, różne style, różne procesy kontroli (wymagania procesowe)

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Wymagania w inżynierii systemów i oprogramowania (Wikipedia)

- W inżynierii systemów wymaganie może być opisem tego co system musi wykonywać w postaci wymagania **funkcjonalnego**
- Inny typ wymagania specyfikuje sam system i jak dobrze wykonuje on swoje funkcje
- Takie wymagania są często nazywane wymaganiami **pozafunkcjonalnymi**, 'wydajnościowymi' lub 'jakościowymi'.
- Przykładami takich wymagań są przydatność, dostępność, niezawodność, podatność na testowanie, możliwości utrzymania oraz (jeśli jest to zdefiniowane w sposób umożliwiający jednoznaczną weryfikację i pomiar) łatwość użytkowania

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Wymagania w inżynierii systemów i oprogramowania (Wikipedia)

- Zbiór wymagań definiuje charakterystyki lub cechy oczekiwanego systemu
- 'Dobra' lista wymagań generalnie nie mówi jak wymagania są implementowane, pozostawiając to projektantowi systemu
- Opis jak system powinien być zrealizowany jest znany jako tendencja implementacji lub "inżynieria rozwiązań". Jednak ograniczenie realizacyjne (znane także jako wymagania rozwiązań) mogą być celowo wyrażone przez przyszłego użytkownika, na przykład w celu zapewnienia zgodności z innymi już posiadanymi produktami

Klasyfikacja wymagań (Wikipedia)

Wymagania są dzielone na następujące kategorie:

- **Wymagania funkcjonalne** opisują funkcje (czynności, operacje, usługi), które system ma realizować, na przykład: zbieranie informacji o klientach firmy lub wprowadzanie/usuwanie/modyfikacja towarów do/z/w koszyka zakupów. Czasami są znane jako możliwości
- **Wymagania pozafunkcjonalne** specyfikują kryteria osądzania działania systemu. Są one znane jako wymagania jakościowe
- **Wymagania ograniczeń** określają granice rozwiązania. Niezależnie od tego jak problem jest rozwiązany, ograniczenia muszą być respektowane

Analiza wymagań funkcjonalnych

- Po wykonaniu identyfikacji wymagań (patrz: Techniki pozyskiwania wymagań) należy przeprowadzić ich analizę celem uszczegółowienia, ewentualnego odrzucenia wymagań sprzecznych lub nieprawidłowych oraz określenia priorytetu i ryzyka
- Analiza wymagań funkcjonalnych umożliwia zidentyfikowanie i opisanie pożądanego zachowania systemu
- Zgodnie z jedną z definicji, wymaganie funkcjonalne to:
 - „stwierdzenie, jakie usługi ma oferować system
 - jak ma reagować na określone dane wejściowe
 - oraz jak ma się zachowywać w określonych sytuacjach
 - w niektórych wypadkach wymagania funkcjonalne określają, czego system nie powinien robić”

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Analiza wymagań funkcjonalnych

- Szczegółowe wymagania funkcjonalne muszą opisywać możliwości systemu w zakresie zachowania oraz dostępnych operacji (czynności wykonywane przez system, odpowiedzi na akcje użytkownika).
- Wymagania funkcjonalne mogą obejmować również pewne elementy dotyczące wyglądu projektowanej aplikacji. Nie jest to jednak regułą, zwykle wymagania GUI znajdują się w osobnej kategorii lub są dostarczane przez Klienta w postaci standardu GUI danej organizacji
- Faza określania wymagań (wyniki) - może być uzupełniona o dodatki:
 - Wymagania sprzętowe
 - Wymagania dotyczące bazy danych
 - Indeks – pomoc w wyszukiwaniu w dokumencie konkretnych informacji

Analiza wymagań funkcjonalnych

Wymagania funkcjonalne powinny opisywać:

- w jaki sposób system realizuje założone cele i wyniki biznesowe w ramach danej dziedziny
- jakie warunki muszą być spełnione, aby system mógł wykonać określone zadania,
- w jaki sposób użytkownik będzie mógł korzystać z systemu w celu realizacji określonych zadań
 - jaki moduł aplikacji oferuje określoną funkcjonalność,
 - jakie czynności użytkownik musi wykonać w celu uzyskania danego rezultatu

Analiza wymagań funkcjonalnych

- Wymagania funkcjonalne powinny uwzględniać standard (np. obowiązujący w danej organizacji), który ma spełniać projektowany system
- Jeśli dotychczasowe rozwiązania wdrożone u Klienta mają określone sposoby realizacji pewnych funkcji, nowe rozwiązanie również powinno je spełniać
- **przykład:**
 - jeśli funkcjonalność wyszukiwania użytkownika już istnieje w którymś z systemów działających u Klienta, w nowym, projektowanym systemie funkcjonalność ta powinna działać analogicznie – zwiększa to re-używalność i spójność aplikacji, ponadto ułatwia pracę programistom i testerom oraz samym użytkownikom końcowym, przyzwyczajonym już do używania danej funkcjonalności w określony sposób

Analiza wymagań funkcjonalnych

- Dobra dokumentacja wymagań nie powinna nadmiernie ograniczać projektu aplikacji, to znaczy narzucać konkretnego rozwiązania architektonicznego
- Analityk powinien w taki sposób opisywać system, by prezentować dostępne funkcje i możliwości aplikacji bez zbędnego wnikania w szczegóły techniczne
- Dlatego tak często mówi się, że analityk nie musi, wręcz nie powinien mieć doświadczenia w programowaniu czy architekturze, gdyż w przeciwnym wypadku mimowolnie będzie się starał napisać dokumentację wymagań w sposób odpowiadający swojej wiedzy technicznej, a to z kolei może znacznie ograniczyć funkcjonalność rozwiązania

Analiza wymagań funkcjonalnych

- Wymagania dokumentuje się zwykle w postaci tekstu.
- Szczegółowy opis wymagań (specyfikacja funkcjonalna) tworzony jest zwykle w postaci tekstu w określonej formie (np. specyfikacja przypadków użycia) oraz modeli UML (np. map procesów lub aktywności).
- Podczas opisywania wymagań funkcjonalnych (zarówno listy wymagań, jak i szczegółowej specyfikacji):
 - Należy posługiwać się prostymi, jednoznacznymi i zrozumiałymi stwierdzeniami
 - zbyt złożone lub zawierające za dużo terminów fachowych powodują, że dokumentacja staje się trudna w odbiorze, a przecież ma służyć nie tylko jej autorowi, lecz również udziałowcom ze strony Klienta, kierownikowi projektu, zespołom developerskim

Analiza wymagań funkcjonalnych

Wady

- **Niejednoznaczność** języka naturalnego; utrudnia precyzyjny zapis wymagań, może prowadzić do różnej interpretacji tego samego tekstu przez różne osoby
- **Elastyczność** języka naturalnego – pozwala te same treści wyrazić na różne sposoby, co utrudnia wykrycie powiązanych wymagań; może prowadzić do przeoczenia sprzeczności wymagań sformułowanych w Różny sposób lecz dotyczących tych samych funkcji.

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Analiza wymagań funkcjonalnych

Dobrze opracowana lista wymagań zawiera wymagania napisane: (1)

- **Prosto** – bez złożonych warunków
- **Spójnie** – terminologia używana w dokumentacji wymagań jest spójna i ma to samo znaczenie w różnych częściach dokumentacji (przykład – słowo „klient” użyte w dokumentacji zawsze znaczy to samo i określa docelowego Klienta z punktu widzenia organizacji zamawiającej produkt IT)
- **Zrozumiale** – analityk nie może zakładać, iż odbiorca dokumentacji posiada szeroką wiedzę dziedzinową. Wymaganie powinno być zrozumiałe zarówno dla eksperta dziedzinowego, testera, programisty, jak i dowolnej innej osoby z grupy udziałowców. Najczęściej popełnianym błędem jest tu zakładanie, że coś jest na tyle oczywiste, iż nie wymaga dokładnego opisu, lub w ogóle nie wymaga wzmianki w dokumentacji. O ile pewne rzeczy są oczywiste dla analityka i przedstawicieli Klienta, o tyle wcale nie muszą być znane programistom, a to spowoduje, że nie będą zaimplementowane
- **Precyzyjnie** – jedna pozycja na liście wymagań dotyczy tylko jednego wymagania. Ułatwia to dalsze śledzenie i monitorowanie implementacji oraz weryfikacji wymagania
- **Zwięźle** – tekst wymagań powinien być w miarę możliwości krótki i powinien opisywać wyłącznie zakres danego wymagania

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Analiza wymagań funkcjonalnych

Dobrze opracowana lista wymagań zawiera wymagania napisane: (2)

- W niektórych przypadkach lista wymagań definiuje również osoby odpowiedzialne za dokumentację określonych wymagań i sposób ich weryfikacji w projektowanym systemie
- Ponadto ważne jest określenie priorytetu i krytyczności dla każdego wymagania
- Wymagania opisane w formie tekstowej powinny obejmować opis możliwości systemu, warunków, jakie muszą zaistnieć, by wymaganie mogło zostać spełnione i zrealizowane oraz wszelkich ograniczeń, jakie uniemożliwiają spełnienie wymagania w projektowanym systemie
- W określonych przypadkach (np. jeśli wymagań jest niewiele i są łatwe do zarządzania) nie ma potrzeby prowadzenia pełnej dokumentacji wymagań – lista (katalog) wymagań wraz z odniesieniami do dokumentacji szczegółowej jest informacją wystarczającą. Przykładowa struktura katalogu wymagań jest przedstawiona w tabeli na kolejnym slajdzie

Analiza wymagań funkcjonalnych

Tabela 1. Przykładowa lista wymagań

ID	Opis	Priorytet	Krytyczność
REQ001	System umożliwia zalogowanie się użytkownika za pomocą loginu i hasła	Wysoki	Wysoki
REQ002	System waliduje poziom uprawnień użytkownika	Wysoki	Średni
REQ003	System blokuje dostęp po 3 nieudanych próbach zalogowania się	Średni	Średni

Klasyfikacja wymagań (Wikipedia)

Wymagania pozafunkcjonalne mogą być dalej klasyfikowane według tego czy są to:

- wymagania użyteczności,
- wymagania wyglądu i odczuwania,
- wymagania humanitarne,
- wymagania wydajnościowe,
- wymagania eksploatacyjne,
- wymagania utrzymywania,
- wymagania bezpieczeństwa,
- wymagania niezawodnościowe,
- lub jeden z wielu innych typów wymagań.

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Klasyfikacja wymagań (Wikipedia)

Wymagania pozafunkcjonalne:

- W inżynierii oprogramowania ta klasyfikacja jest użyteczna, gdyż tylko wymagania funkcjonalne są bezpośrednio implementowane w programach.
- Wymagania pozafunkcjonalne są kontrolowane przez inne aspekty systemu, przykład:
 - **niezawodność** systemu komputerowego zależy od ilości usterek sprzętu,
 - **wydajność** zależy od wydajności procesora i pamięci.
- Wymagania pozafunkcjonalne mogą być, w pewnych przypadkach, dekomponowane na jedno lub kilka wymagań funkcjonalnych.
- Dodatkowo wymagania pozafunkcjonalne, nie posiadające miary, mogą być przekształcane w wymagania procesowe. Na przykład możliwości utrzymania systemu mogą być dekomponowane na ograniczenia składników systemu lub na liczbę wierszy kodu programu.

Klasyfikacja wymagań

Wymagania pozafunkcjonalne:

- Wymagania pozafunkcjonalne: nie dotyczą bezpośrednio konkretnych funkcji systemu: ograniczenia usług i funkcji systemu (np. czasowe, dotyczące procesu tworzenia, standardy, itd.).
- Mogą być związane z właściwościami systemu, jak: czas reakcji, niezawodność, zajętość pamięci.
- Mogą definiować ograniczenia systemu (możliwości urządzeń wejścia-wyjścia, reprezentacje danych używane przez interfejsy).

Klasyfikacja wymagań

Wymagania pozafunkcjonalne wynikają z:

- potrzeb użytkownika
- ograniczeń budżetowych
- strategii firmy
- konieczności współpracy z innymi systemami (sprzętu lub oprogramowania)
- czynników zewnętrznych jak przepisy o bezpieczeństwie, ustawy chroniące prywatność, itd.

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Klasyfikacja wymagań pozafunkcjonalnych:

- **Wymagania produktowe** – zachowanie produktu (wymagania efektywnościowe – szybkość działania systemu i potrzeb pamięci, wymagania niezawodności – akceptowalna częstość awarii), wymagania przenośności, wymagania użyteczności.
- **Wymagania organizacyjne** – wynikają ze strategii i procedur w firmie-kliencie i w firmie-wytwórcy (standardy procesu, wymagania implementacyjne: język, metoda programowania, wymagania dostawy – kiedy dostarczyć produkt i jego dokumentacje).
- **Wymagania zewnętrzne** – szeroka kategoria: wszystkie wymagania wynikające z czynników zewnętrznych dla systemu i procesu jego tworzenia (wymagania współpracy: interakcje z systemami innych firm, prawne, etyczne: akceptacja).

Klasyfikacja wymagań pozafunkcyjnych:

Weryfikowalność niektóre właściwości – wyrażone przez miary, możliwość testowania obiektywnego, np.:

- **szybkość** (liczba transakcji/sek, czas reakcji, czas odświeżania),
- **rozmiar** (KB, MB),
- **łatwość użycia** (czas szkolenia, liczba okien pomocy),
- **niezawodność** (średni czas bez awarii, częstość błędów),
- **solidność** (czas uruchomienia po awarii, prawdopodobieństwo utraty danych, procent zdarzeń powodujących awarie),
- **przenośność** (procent poleceń zależnych od platformy, liczba docelowych systemów).

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Weryfikacja wymagań *(Wikipedia)*

- Wszystkie wymagania powinny być weryfikowalne
- Najpowszechniejszą metodą weryfikacji jest badanie
- Jeśli nie ma to zastosowania inna metoda może być użyta (analiza, pokaz, oględziny lub recenzja projektu)
- Pewne wymagania są nieweryfikowalne z powodu ich struktury. Obejmuje to wymagania mówiące, że system „nigdy” lub „zawsze” będzie miał określoną właściwość
- Odpowiednie przebadanie tych wymagań może potrzebować nieokreślonego cyklu testowania
- Takie wymagania muszą być sformułowane ponownie, aby były weryfikowalne
- Zgodnie z powyższymi stwierdzeniami wszystkie wymagania muszą być weryfikowalne

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Weryfikacja wymagań *(Wikipedia)*

- Wymagania pozafunkcjonalne, nieweryfikowalne na poziomie programu:
 - muszą być zachowane jako dokumentacja intencji klienta
 - mogą być przekształcone na wymagania procesowe jako sposób ich praktycznego wypełnienia
- Inne wymagania pozafunkcjonalne mogą prowadzić do składników systemu i weryfikacji na tym poziomie
- Na przykład niezawodność systemu jest często weryfikowana przez analizy na poziomie składników systemu

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Weryfikacja wymagań *(Wikipedia)*

- Wymagania mogą być niejednoznaczne, niekompletne i niespójne
- Są znane techniki, które pomagają uporać się z tymi niedoskonałościami
- Usunięcie niejednoznaczności, niekompletności i braku spójności na etapie tworzenia wymagań kosztuje o rząd wielkości mniej niż ich poprawianie w późniejszych fazach rozwijania produktu
- Analiza wymagań temu służy.
- Przeciwnieństwem niejasności wymagań jest ich wielka szczegółowość, która prowadzi do tego, że:
 - ich wytworzenie zajmuje dużo czasu,
 - ograniczają dostępne możliwości realizacji,
 - ich wytworzenie jest kosztowne.
- Analiza wymagań pod tym kątem pozwala na inżynierski kompromis.

Weryfikacja wymagań pozafunkcyjnych

Dobrze sformułowane wymagania pozafunkcyjne:

- powinny być weryfikowalne – możliwość sprawdzenia czy system je spełnia.
- wymagania sformułowane jako stwierdzenia, np.: „*łatwy w obsłudze*”, „*niezawodny*”, „*wydajny*” - nie mogą być obiektywnie zweryfikowane
- konieczność wyrażenia wymagań przy pomocy wielkości mierzalnych

Weryfikacja wymagań pozafunkcyjnych (1)

Przykład formularza opisu wymagań pozafunkcyjnych, np. w aspekcie rozważanych rozwiązań:

Cecha:

- Wydajność:
- Rozmiar:
- Łatwość użytkowania:

Miary:

Liczba trans. obsługowanych/s.
Wymagana pamięć RAM
Wymagana pamięć dyskowa
Czas dla przeszkolenia
Liczba stron dokumentacji

Weryfikacja wymagań pozafunkcyjnych (2)

Przykład formularza opisu wymagań pozafunkcyjnych, np. w aspekcie rozważanych rozwiązań:

<u>Cecha</u>	<u>Miary</u>
Niezawodność	
....	Prawdopodobieństwo błędnego wykonania podczas realizacji transakcji
...	Częstotliwość błędnych wykonań
...	Średni czas między błędnymi wykonaniami
...	Dostępność (procent czasu)

Zajęcia nr 3 – Identyfikacja dziedziny problemu

Weryfikacja wymagań pozafunkcyjnych (3)

Przykład formularza opisu wymagań pozafunkcyjnych, np. w aspekcie rozważanych rozwiązań:

<u>Cecha:</u>	<u>Miary:</u>
<i>Odporność</i>	
....	Czas restartu po awarii systemu
...	Prawdopodobieństwo zniszczenia danych w przypadku awarii

<u>Cecha:</u>	<u>Miary:</u>
<i>Przenośność</i>	
....	Procent kodu zależnego od platformy docelowej
....	Liczba platform docelowych
...	Koszt przeniesienia na nową platformę

Dokumentacja wymagań (Wikipedia)

- Wymagania są zwykle pisane w celu komunikacji pomiędzy różnymi interesariuszami
- Oznacza to, że wymagania powinny być łatwe do zrozumienia zarówno przez użytkowników jak i projektantów
- Drogą wspólnego dokumentowania wymagań jest stwierdzenie co system będzie robił
- Przykład: „Dostawca ma dostarczyć produkt nie później niż dnia xyz.”
- Innymi możliwościami są przypadki użycia

Zmiany wymagań *(Wikipedia)*

- Generalnie wymagania zmieniają się z czasem
- Wymagania zdefiniowane i zatwierdzone powinny podlegać kontroli zmian
- W wielu projektach wymagania były zmieniane przed ukończeniem systemu
- Częściowo wynika to ze złożoności oprogramowania i faktu, że *„użytkownicy nie wiedzą co chcą zanim tego nie zobaczą”*
- Te charakterystyki wymagań podlegają zarządzaniu wymaganiami

1 Przykład analizy – Warsztat samochodowy:

1. Istnieje (już funkcjonuje) warsztat samochodowy
2. Klient przyjeżdża samochodem z usterką lub usterkami
3. Samochód jest naprawiany przez mechanika lub mechaników
4. Po naprawie klient płaci za usługę

Czego nam w tym opisie brakuje?

•Analiza danych:

- kto to jest klient ?
- co to jest samochód?
- Co to jest usterka
- Kto to jest mechanik?
- Co to jest naprawa?
- Co to jest zapłata?
- Zależności między nimi? (jeden do wielu, wiele do wielu)
- Jakie informacje (wiedzę zapisaną w bazie) będziemy (potrzebujemy w przyszłości) analizować, raportować ?
- **Uwaga:** W analizie nie mówimy o tabelach?

Zajęcia nr 2 - Analiza Zadania Dyplomowego

2 Przykład analizy – Port jachtowy:

1. Istnieje port jachtowy na jeziorze X
2. Port posiada miejsca do cumowania jachtów
3. Port posiada kilka pomostów < pomost A, B, C...>, przy których są miejsca do cumowania <miejsce 1, 2, ... 10.. > (uwaga: przy jednym pomoście może być wiele miejsc), miejsce ma określoną długość < 10m, 15 m, 20 m>, potrzebna jest również wiedza czy miejsce jest aktualnie wolne czy zajęte, niedostępne bo awaria, naprawa
4. W porcie cumują jachty, potrzebna wiedza <nazwa jachtu, długość jachtu, właściciel jachtu>
5. Potrzeba wiedzy o właścicielu jachtu < imię, nazwisko, adres, kraj> (uwaga: właściciel może mieć wiele jachtów, jeden jacht <cumujący> ma jednego właściciela).
6. Zajęcie miejsca cumowania <parkowanie> to: jaki jacht, jaki właściciel, które zajął miejsce na przystani, data przyplłynięcia, data odpłynięcia, łączna cena za cumowanie za dobę w tym miejscu.
7. Cumując można korzystać z pewnych (dodatkowo płatnych) usług <wymycie pokładu, doładowanie akumulatorów> jakie potrzebne informacje oprócz cennika?
8. Potrzeba ewidencjonowania tych usług, (uwaga jedno cumowanie może być wiele usług),
9. Cumując można odpłatnie pożyczyć na przystani narzędzia <młotek, siekiera, wiertarka> potrzeba ewidencjonowania tych płatnych pożyczek (uwaga jedno cumowanie może być wiele pożyczek sprzętu).

Opracować część analityczną pracy dyplomowej w zakresie:

- Wymagania funkcjonalne systemu*
- Wymagania pozafunkcjonalne systemu*
- Użytkownicy systemu i dostępne im funkcje*

Seminarium Dyplomowe

Dziękuję!!!!!! Tyle na dzisiaj

