

Zastosowanie informatyki w gospodarce - projekt

System obsługi szpitala

AUTORZY:

inż. Jakub Celadyn, inż. Marcin Jackowski, inż.
Przemysław Kaśków, inż. Tomasz Papiór, inż.
Mateusz Sobala

PROWADZĄCY PRACĘ:

Dr inż. Marek Woda

OCENA PRACY:

Spis treści

1. Temat projektu.....	3
2. Zakres projektu.....	4
2.1. Cele.....	4
2.2. Ryzyka	4
2.3. Funkcjonalności	5
3. Narzędzia i technologie zastosowane projekcie	13
4. Aktualny stan rynku	15
5. Kamienie milowe i stan projektu.....	16
5.1. Wykres Gaunta	16
5.2. Kalkulacje kosztów i nakład pracy	16
6. Studium wykonywalności	17
7. Opis implementacji i wdrożenia projektu	18
8. Wnioski	21
Załącznik nr 1 Schemat bazy danych.....	22

1. Temat projektu

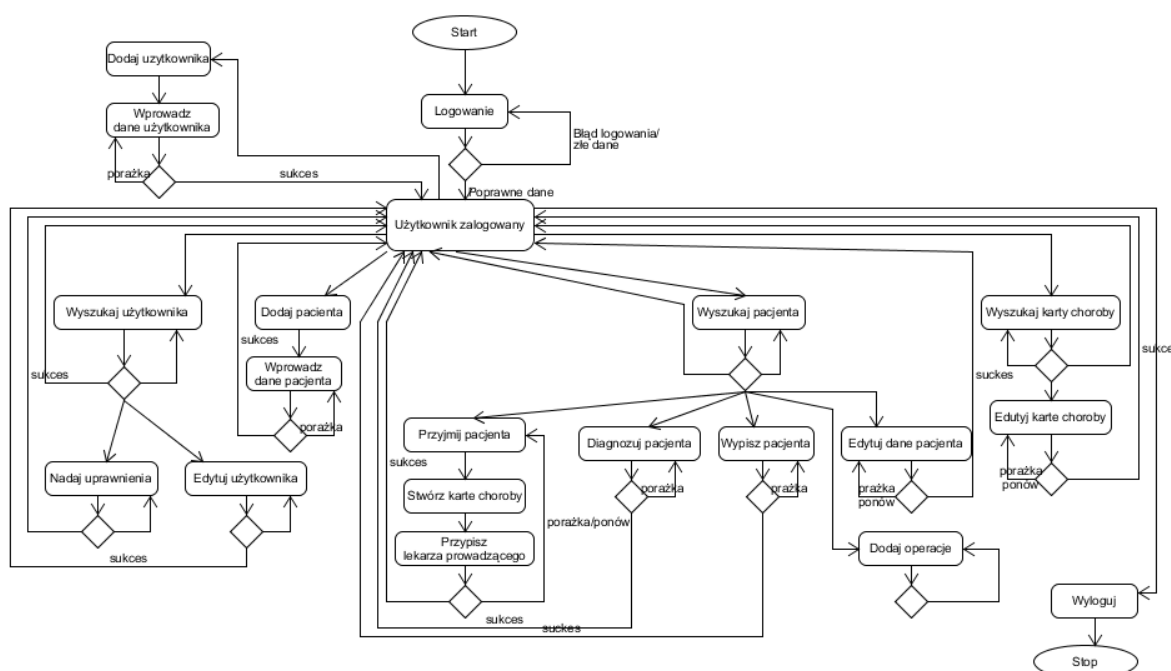
Pomysł na temat projektu został zaproponowany przez członka grupy, Marcina Jackowskiego. Genezą pomysłu jest styczeń z dużym czasem oczekiwania na jakąkolwiek akcję bądź informację podczas pobytu w szpitalu oraz w trakcie przyjmowania do szpitala. Chęć zoptymalizowania procesu obsługi pacjenta w szpitalu miała największy wpływ na wybór tego tematu. Ponad to jest to dobra okazja do zgłębienia zagadnień informatyki stosowanej w gospodarce oraz w medycynie, a nawet częściowo w telemedycynie.

System obsługi szpitala, sam w sobie jako system informatyczny nie jest innowacją. Na polskim rynku można znaleźć obecnie co najmniej trzy firmy proponujące taką usługę. Innowacyjność naszego projektu ma polegać na mobilności dostępu do kart pacjenta z prostym interfejsem, za cenę, która nawet małym ośrodkom pozwala na jego wdrożenie.

2. Zakres projektu

2.1. Cele

Głównym celem projektu jest stworzenie kompletnego systemu obsługującego działanie szpitala. Niestety czas przeznaczony na projekt uniemożliwia stworzenie takiego systemu w ciągu jednego semestru. Dlatego też celem projektu w trakcie obecnego semestru jest wykonanie części białej systemu obsługi szpitala, a dokładniej części odpowiedzialnej za obsługę pacjenta w jednostce medycznej. Czyli wykonanie elektronicznego obiegu informacji o pacjencie, zapewnienie mobilnego dostępu do kart chorobowych pacjentów dla pielęgniarek, zapisywanie wykonanych czynności w szpitalu w karcie chorobowej pacjenta (np. godz. podania leku, wizyta lekarza), przeglądanie historii chorób pacjentów, przeglądanie pacjentów z daną chorobą



Rysunek 2.1 Schemat blokowy programu

2.2. Ryzyka

Wykonywanie jakiegokolwiek projektu wiąże się z ryzykiem. Ryzyko projektu to nic innego jak możliwość jego porażki. Ze względu na przygotowanie w projekcie modułu składającego się na większy system największy wpływ na brak sukcesu ma jakość pracy grupy i jakość wykonania modułu. Tabela 2.1 przedstawia zestawienie czynników mogący mieć negatywny wpływ na projekt.

Tabela 2.1 Ryzyka projektu

Rodzaj ryzyka	Szansa wystąpienia	Wpływ na porażkę
nieumiejętność pracy w zespole	0,05	0,5
nieumiejętność wykorzystania narzędzi	0,03	0,45

niespójność danych	0,01	0,9
nieczytelność danych	0,1	0,9
nieuzasadniona złożoność rozwiązań	0,25	0,19
ograniczenie zakresu funkcji	0,1	0,9
nieumiejętność planowania	0,2	0,95

Dane liczbowe w tabeli są przedstawione w skali od 0 do 1, gdzie 0 oznacza niemożliwość wystąpienia ryzyka bądź brak wpływu na porażkę, zaś 1 oznacza pewność wystąpienia ryzyka, bądź że ryzyko spowoduje pewną porażkę projektu.

2.3. Funkcjonalności

Id wymagania	1
Nazwa	Logowanie
Opis	Zalogowanie użytkownika do systemu
Przesłanka	Zalogowanie użytkownika do systemu i jednoznaczne zidentyfikowanie go
Dane wejściowe	Dane niezbędne o przeprowadzenia poprawnej weryfikacji podczas logowania
Wynik	Zalogowanie użytkownika do systemu i udostępnienie mu określonych funkcji
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	2
Nazwa	Dodaj nowego pacjenta
Opis	Dodanie nowego pacjenta do systemu
Przesłanka	Dodanie nowego pacjenta do systemu i nadanie mu unikatowego kodu identyfikacyjnego
Dane wejściowe	Dane niezbędne do jednoznacznego zidentyfikowania konkretnego pacjenta wraz z danymi kontaktowymi i historia choroby
Wynik	Dodanie nowego pacjenta opisanego konkretnymi danymi
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	3
Nazwa	Dodaj kartę choroby
Opis	Dodanie nowej karty choroby
Przesłanka	Dodanie nowej karty choroby powiązanej z konkretnym pacjentem
Dane wejściowe	Dane uzyskane przez lekarza w wyniku przeprowadzenia badania lekarskiego
Wynik	Dodanie nowej karty choroby z danymi uzyskanymi przez lekarza
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem

Priorytet	Wymagane
-----------	----------

Id wymagania	4
Nazwa	Edytuj kartę choroby
Opis	Zmiana danych w karcie choroby
Przesłanka	Modyfikacja danych opisujących stan pacjenta
Dane wejściowe	Dane, które muszą zostać zmienione, odnoszącego się do konkretnego pacjenta
Wynik	Zmodyfikowanie danych opisujących stan konkretnego pacjenta
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	5
Nazwa	Archiwizuj kartę choroby
Opis	Przeniesienia karty choroby do archiwum
Przesłanka	Nadanie karcie choroby statusu archiwalnego
Dane wejściowe	Karta choroby
Wynik	Karta choroby ze statusem archiwalnym
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	6
Nazwa	Edytuj dane pacjenta
Opis	Zmiana danych pacjenta
Przesłanka	Modyfikacja danych opisujących konkretnego pacjenta
Dane wejściowe	Dane, które muszą zostać zmienione, odnoszące się do konkretnego pacjenta
Wynik	Zmodyfikowanie danych opisujących konkretnego pacjenta
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	7
Nazwa	Przeglądaj karty choroby
Opis	Przeglądanie kart choroby pacjentów szpitala
Przesłanka	Umożliwienie przeglądania i sprawdzania danych zawartych w kartach pacjentów
Dane wejściowe	Kryteria wyszukiwania
Wynik	Prezentacja kart choroby spełniających kryteria wyszukiwania
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	8
Nazwa	Przyjmij pacjenta do szpitala

Opis	Dodanie nowego pobytu pacjenta w szpitalu
Przesłanka	Możliwość przyjęcia pacjenta do szpitala
Dane wejściowe	Dane niezbędne do jednoznacznej identyfikacji konkretnego pacjenta
Wynik	Przyjęcie pacjenta do szpitala
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	9
Nazwa	Wypisz ze szpitala
Opis	Zakończenie pobytu pacjenta w szpitalu
Przesłanka	Umożliwienie wypisania pacjenta ze szpitala
Dane wejściowe	Dane niezbędne do jednoznacznej identyfikacji konkretnego pacjenta i aktualnego pobytu w szpitalu
Wynik	Wypisanie pacjenta ze szpitala
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	10
Nazwa	Edycja konto użytkownika
Opis	Modyfikacja danych opisujących konkretnego użytkownika
Przesłanka	Umożliwienie edycji konta użytkownikowi
Dane wejściowe	Dane, które muszą zostać zmienione, odnoszące się do konkretnego użytkownika
Wynik	Zmodyfikowanie danych opisujących konkretnego użytkownika
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	11
Nazwa	Przypisz lekarza prowadzącego do pacjenta
Opis	Dodanie lekarza prowadzącego w karcie choroby pacjenta
Przesłanka	Określenie lekarza prowadzącego pacjenta podczas pobytu w szpitalu
Dane wejściowe	Dane niezbędne do jednoznacznej identyfikacji konkretnego pacjenta wraz z jego aktualną kartą choroby oraz identyfikacja konkretnego pacjenta
Wynik	Nadanie pacjentowi lekarza prowadzącego
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	12
Nazwa	Generuj Qr code
Opis	Generacja Qr code
Przesłanka	Wygenerowanie Qr code dla pacjenta

Dane wejściowe	Dane niezbędne do identyfikacji konkretnego pacjenta
Wynik	Wygenerowanie Qr code identyfikującego konkretnego pacjenta
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	13
Nazwa	Skanuj Qr code
Opis	Zeskanowanie Qr code
Przesłanka	Zeskanowanie Qr code w celu zidentyfikowania pacjenta
Dane wejściowe	Qr code
Wynik	Otrzymanie danych identyfikujących konkretnego pacjenta
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	14
Nazwa	Diagnostuj pacjenta
Opis	Wprowadź diagnozę pacjenta
Przesłanka	Wprowadzenie diagnozy pacjenta przeprowadzoną przez lekarza
Dane wejściowe	Dane identyfikujące konkretnego lekarza i pacjenta, treść diagnozy i kod
Wynik	Dodanie diagnozy do karty choroby pacjenta
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	15
Nazwa	Dodaj użytkowników
Opis	Dodanie użytkownika do systemu
Przesłanka	Dodanie użytkownika do systemu obsługi szpitala
Dane wejściowe	Dane niezbędne do identyfikacji konkretnego lekarza
Wynik	Dodanie nowego użytkownika opisanego konkretnymi danymi
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	16
Nazwa	Nadaj uprawnienia
Opis	Nadanie uprawnień użytkownikowi
Przesłanka	Nadanie uprawnień użytkownikowi odzwierciedlające jego uprawnienia w szpitalu
Dane wejściowe	Dane niezbędne do identyfikacji konkretnego użytkownika oraz rodzaj uprawnień
Wynik	Nadanie uprawnień konkretnemu użytkownikowi
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem

Priorytet	Wymagane
-----------	----------

Id wymagania	17
Nazwa	Dodaj operacje
Opis	Dodanie operacji do systemu
Przesłanka	Dodanie operacji do systemu przez lekarza
Dane wejściowe	Dane identyfikujące konkretnego pacjenta, typ operacji oraz miejsce
Wynik	Dodanie operacji w systemie
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Wymagane

Id wymagania	18
Nazwa	Skanuj dokument
Opis	Zeskanowanie dokumentu
Przesłanka	Zeskanowanie dokumentu w celu zamieszczenia w systemie i powiązania z konkretnym pacjentem
Dane wejściowe	Zeskanowany dokument oraz dane niezbędne do zidentyfikowania konkretnego pacjenta
Wynik	Dodanie dokumentu do danych pacjenta/ karty choroby
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

Id wymagania	19
Nazwa	Generuj dokument
Opis	Wygenerowanie dokumentu
Przesłanka	Wygenerowanie dokumentu dla pacjenta przebywającego w szpitalu
Dane wejściowe	Dane niezbędne do wygenerowania dokumentu oraz identyfikator typu dokumentu
Wynik	Wygenerowanie dokumentu z podanymi danymi
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

Id wymagania	20
Nazwa	Drukuj dokument
Opis	Wydrukowanie dokumentu
Przesłanka	Wydrukowanie dokumentu wygenerowanego w systemie
Dane wejściowe	Wygenerowany dokument
Wynik	Wydrukowanie dokumentu
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

Id wymagania	21
Nazwa	Generuj grafik pracy

Opis	Wygenerowanie grafiku pracy
Przesłanka	Wygenerowanie grafiku pracy zawierającego informację o godzinach pracy pracowników szpitala
Dane wejściowe	Dane niezbędne do konkretniej identyfikacji konkretnego pracownika oraz godziny jego pracy
Wynik	Wygenerowanie grafiku pracy pracowników szpitala w formie grafiki prezentującej kalendarz wraz okresami pracy pracowników
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

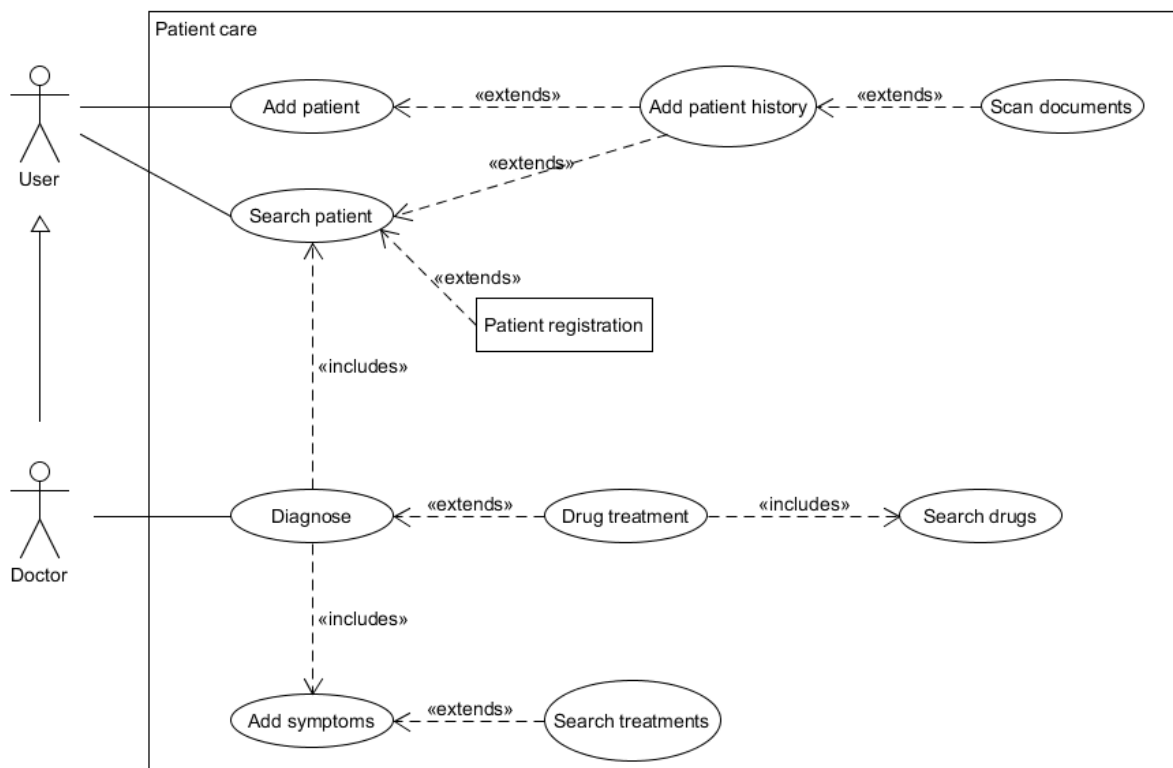
Id wymagania	22
Nazwa	Zgłoś urlop
Opis	Zgłoszenie wniosku o urlop
Przesłanka	Umożliwienie zgłoszenia wniosków o urlop w formie elektronicznej
Dane wejściowe	Dane niezbędne do identyfikacji konkretnego pracownika oraz daty określające początek i koniec urlopu
Wynik	Zgłoszeni urlopu w systemie
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

Id wymagania	23
Nazwa	Akceptuj wniosek o urlop
Opis	Akceptacja wniosku o urlop
Przesłanka	Przyznanie urlopu pracownikowi szpitala
Dane wejściowe	Wniosek o urlop
Wynik	Akceptacja urlopu
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

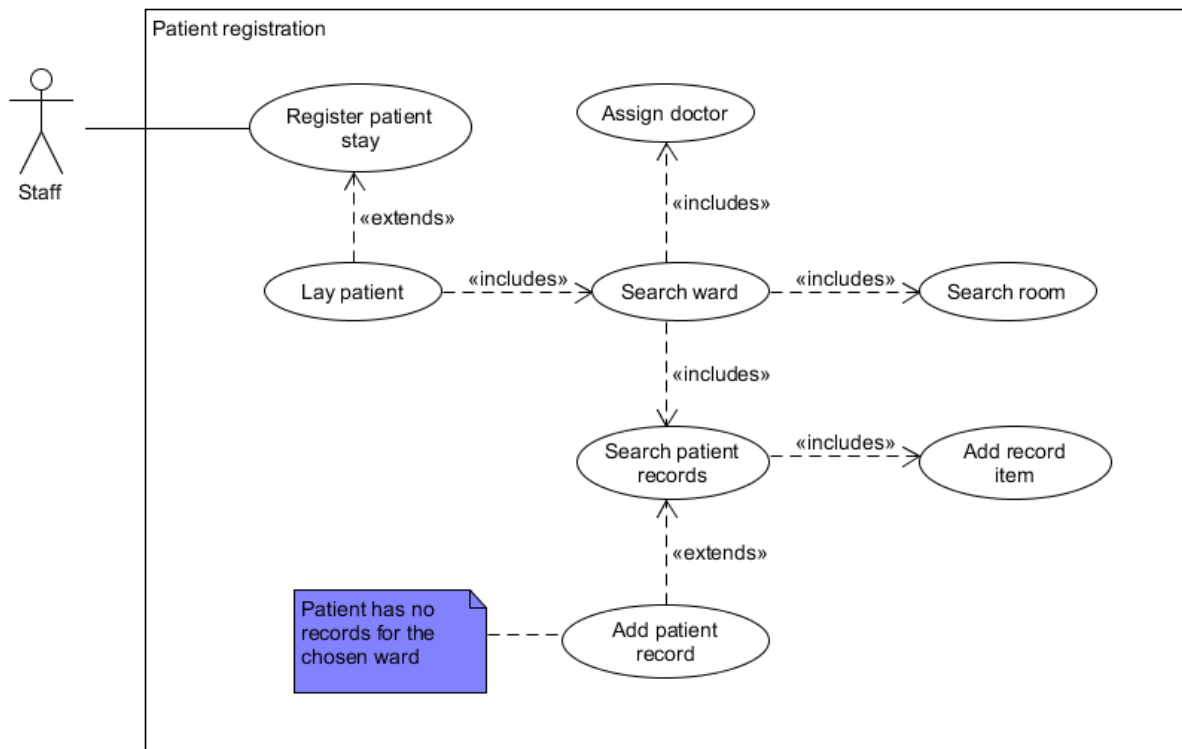
Id wymagania	24
Nazwa	Odrzuć wniosek o urlop
Opis	Odrzucenie wniosku o urlop
Przesłanka	Nie wyrażenie zgodny na urlop dla pracownika szpitala
Dane wejściowe	Wniosek o urlop
Wynik	Odrzucenie wniosku urlopowego
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)

Id wymagania	25
Nazwa	Monitoruj stan zdrowia pacjenta
Opis	Monitorowanie stanu zdrowia pacjenta online
Przesłanka	Umożliwienie monitorowania funkcji życiowych pacjenta online

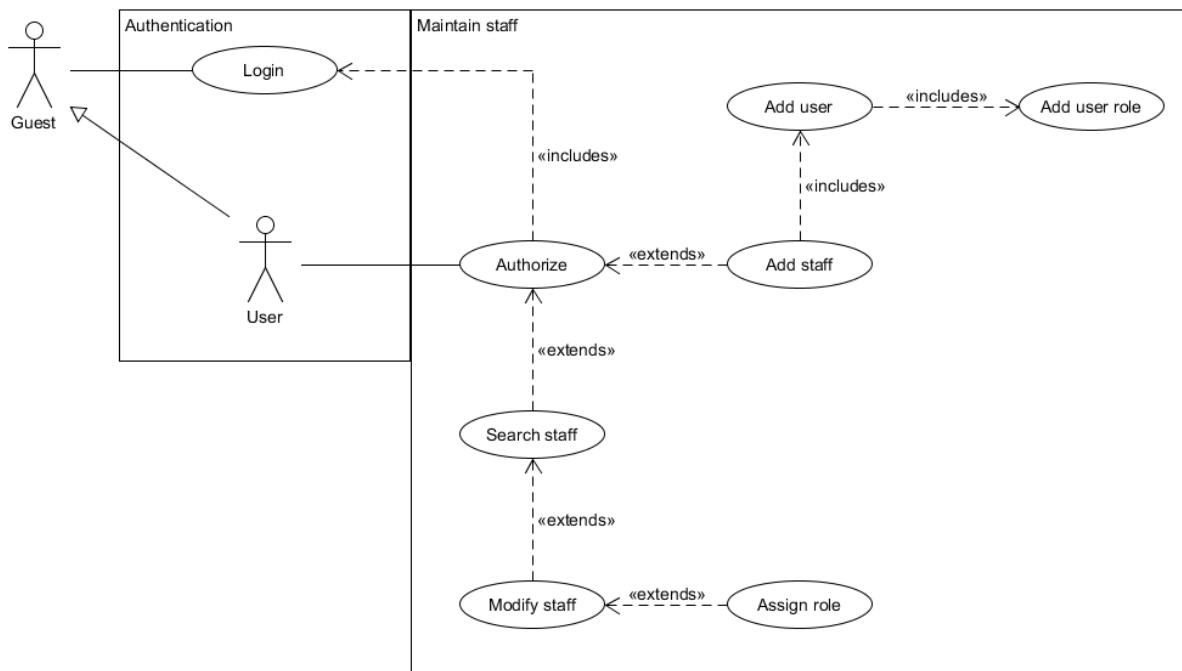
Dane wejściowe	Dane identyfikujące konkretnego pacjenta oraz dane z urządzeń monitorujących funkcje życiowe pacjenta
Wynik	Uzyskanie wizualizacji danych z urządzeń monitorujących konkretnego pacjenta
Kryterium spełnienia	Funkcja zaimplementowana zgodnie z opisem
Priorytet	Rozszerzone (opcjonalne)



Rysunek 2.2 Przypadek użycia dla pacjenta



Rysunek 2.3 Przypadek użycia dla rejestracji



Rysunek 2.4 Przypadek użycia dla zarządzania użytkownikami

3. Narzędzia i technologie zastosowane projekcie

Tabela 3.1 Zestawienie wykorzystanych technologii i narzędzi

Serwer	Raspbin, nginx, Fail2ban, phpMyAdmin
Baza danych	MySQL Workbench, PHP, phpMyAdmin, UMLet
Aplikacja internetowa	Bootstrap, HTML, JavaScript, PHP, phpMyAdmin
Aplikacja mobilna	Xcode, Swift

Aplikacja internetowa i baza danych działa na Raspberry Pi. Jest to platforma komputerowa stworzona przez Raspberry Pi Foundation. Budowa jest oparta o układ Broadcom BCM2835 SoC. W skład układu wchodzi procesor ARM1176JZF-S 700 MHz, procesora graficznego VideoCore IV. Zainstalowany został na nim Raspbina jako system operacyjny. Oparty jest on na Debianie. Został stworzony specjalnie w celu optymalizacji pracy układu Raspberry Pi.

Serwer aplikacji został stworzony przy użyciu nginx. Jest to narzędzie do tworzenia serwera http. Umożliwi również stworzenie serwera mailowego oraz serwera generic TCP proxy. Do kontroli ruchu na serwerze wykorzystano Fail2ban. Głównym jego zadaniem jest ochrona serwera przed atakami brutalnymi. Jego działanie jest oparte na monitorowaniu logów serwera i wykonywanie na nich skryptów w celu wykrycia nieprawidłowości. Umożliwia blokowanie adresów IP, które np.: próbują obejść system zabezpieczeń, wykonują częste próby logowania, wykonują niechciane przez nas akcje.

Baza danych została stworzona przy wykorzystaniu MySQL Workbench. Jest to narzędzie wizualne do tworzenia architektury bazodanowej. Działa w oparciu o strukturalny język zapytań SQL (Structured Query Language). Uważany jest za standard i bardzo powszechny w zastawianiu. Charakteryzuje się prostą i intuicyjną składnią, której nauka jest szybka. Ze względu na dużą powszechność istnieje wiele forów i stron internetowych poświęconych tematyce języka oferujących pomoc w rozwiązywaniu problemów.

Do zarządzania bazą danych poprzez sieć wykorzystano narzędzie phpMyAdmin napisane w języku PHP. PHP jest obiektowym językiem programowania zaprojektowany do generowania internetowych i budowania aplikacji webowych. Jest bardzo popularny i najczęściej stosowany to tworzenia skryptów po stronie serwera. Duża popularność wpływa powszechność występowania różnego rodzaju tutoriali i szeroko zakres pomocy ze strony innych użytkowników. Język PHP został wykorzystany do oskryptowania aplikacji internetowej oraz komunikacji z aplikacją mobilną.

Bootstrap jest darmowym i open-sorce'owym zbiorem narzędzi do tworzenia aplikacji internetowych. Zawiera wzorce dla m.in. HTML, CSS, JavaScript. HTML to hipertekstowy język znaczników, który jest szeroko wykorzystywany do tworzenia stron internetowych. Pozwala opisać strukturę informacji zawartych na stronie internetowej, nadając znaczenie poszczególnym fragmentom tekstu. Składnia opiera się o znaczniki umieszczone w nawiasach trójkątnych (ostrokątnych). Występują w

parach oznaczają początek i koniec, gdzie znacznik końcowy poprzedzony jest „/”. JavaScript jest skryptowym językiem programowania. Stosowany jest do tworzenia stron interaktywnych reagując na zdarzenia, sprawdzania poprawności formularzy lub budowania elementów nawigacyjnych.

Aplikacja mobilna została stworzona przy wykorzystaniu zintegrowanego środowiska programistycznego jakim jest Xcode firmy Apple Inc. Służy do tworzenia aplikacji i oprogramowania na systemy operacyjne firmy Apple. Zapewnia również możliwość debugowania stworzonych projektów. Do napisania aplikacji posłużono się językiem programowania Swift pozwalającym na tworzenie oprogramowania na urządzenia mobilne z systemem iOS. Dużą zaletą jest współpraca kodu Swift'a z Objective-C. Kod taki charakteryzuje się również szybką kompilacją co jest ważną cechą w przypadku urządzeń mobilnych.

Diagramy UML zostały wykonane przy użyciu programu UMLet. Darmowe i open-source'owe narzędzie pozwalające na szybkie rysowanie diagramów. Dodatkowo interfejs wykorzystuje kod formatowania tekstu celu modyfikacji podstawowych kształtów.

4. Aktualny stan rynku

Niestety na rynku panuje duża konkurencja w zakresie systemów obsługi szpitala. Rozwiązania takie proponują nam m.in. Asseco Poland S.A., Kamssoft S.A., Politechnika Poznańska.

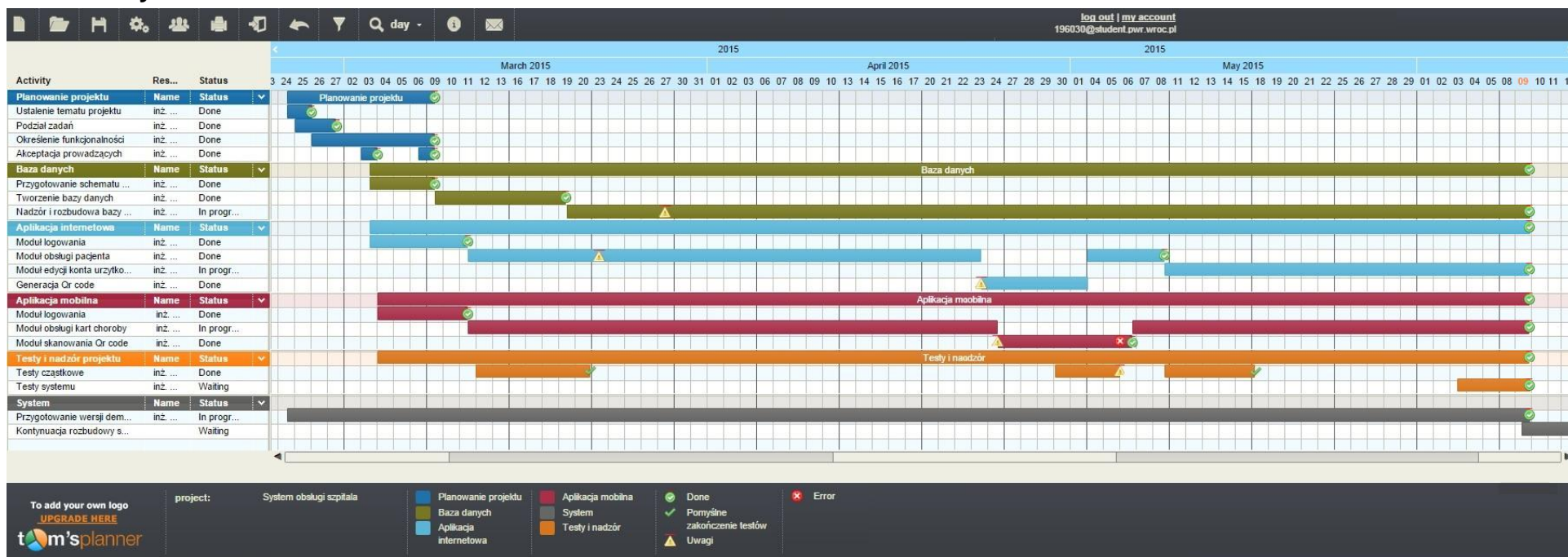
Firma Asseco jest potentatem rynku IT. Jest to najwyższej notowana polska firma informatyczna na Giełdzie Papierów Wartościowych w Warszawie. Obsługuje 450 największych szpitali w Polsce oraz większość Centrów Krwiodawstwa i Krwiolecznictwa. Ich system zapewnia obsługę części medycznej (HIS), obsługę części administracyjnej (ERP), obsługę jednostek świadczących usługi opłacane przez NFZ i komercyjne, symulator JGP, system powiadamiania pacjentów - e-mail i sms, obsługa Elektronicznej Dokumentacji Medycznej, kontrola rozliczeń z NFZ, rejestracja internetowa, obsługa kodów 2D, elektroniczna inwentaryzacja, elektroniczną identyfikacją pacjenta, obsługę zleceń przez Internet.

Kamssoft wyróżnia się autorskim systemem do analizy dystrybucji leków, pozwalający na śledzenie oraz predykcję trendów zachodzących na rynku farmaceutycznym. W swoim systemie medycznym zapewniają m.in. zarządzanie informacją (sprawne gromadzenie, bezpieczne przechowywanie danych, wielowymiarowe analizy oraz statystyki), zarządzanie dokumentacją (formularze medyczne, archiwizacja dokumentacji, automatyczne generowanie wydruków), zarządzanie kosztami (analiza kosztów, optymalizacja kosztów leczenia), zarządzanie świadczeniami oraz lekami (elektroniczny przepływ informacji, farmakoekonomika, rejestracja świadczeń medycznych, zarządzanie pobytem pacjenta w szpitalu), zarządzanie strumieniem pracy (organizacja pobytów pacjentów oraz realizacji świadczeń), zarządzanie czasem (dostęp do pełnej informacji w każdej chwili, elektroniczny przepływ informacji w ramach instalacji sieciowej).

Podobnie jak my obecnie w ramach projektu, tak również pracownicy Politechniki Poznańskiej tworzyli system obsługi szpitala. Duży nakład pracy osób z wieloletnim doświadczeniem zaowocował pierwszym wdrożeniem systemu w 1996 roku. Dzięki rozwojowi systemu obsługuje on już ponad 200 placówek w Polsce. Obecnie rozwojem systemu zajmuje się firma Rightsoft, która powstała poprzez wydzielenie się projektu z Politechniką Poznańską.

5. Kamienie milowe i stan projektu

5.1. Wykres Gaunta



Rysunek 5.1 Wykres Gaunta

5.2. Kalkulacje kosztów i nakład pracy

Tabela 5.1 Zestawienie kosztów i nakładu pracy

Rodzaj kosztów	Kwota [PLN]	Rodzaj kosztów	Kwota [PLN]
200h x 20PLN/h x 5osób	20 000	Zysk	0
Ubezpieczenie (19,64%)	3 928	VAT (23%)	6 604
Koszty bezpośrednie	23 928	Koszt projektu	35 316
Koszty pośrednie (20%)	4 785		
Suma Kosztów	28 713		

6. Studium wykonywalności

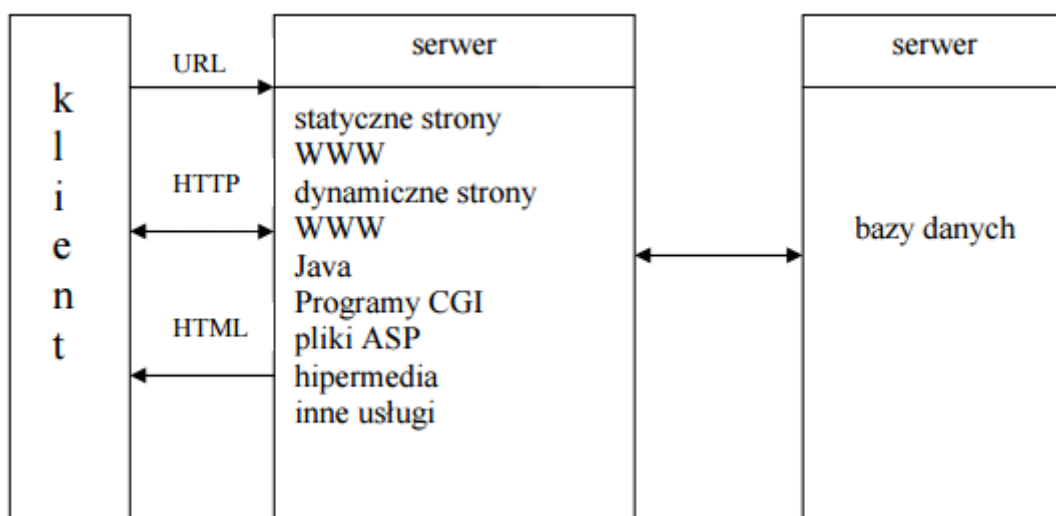
Projekt przebiegał bez konfliktów i terminowo. Udało się wykonać wszystkie zamierzone funkcjonalności systemu. Posiadając kolejny miesiąc możliwe byłoby stworzenie zarządzania czasem pracy oraz urlopami pracowników szpitala. W ciągu dodatkowych czterech miesięcy możliwe byłoby stworzenie obiegu wyników pacjenta bezpośrednio z urządzeń pomiarowych oraz śledzenie stanu zdrowia pacjenta online. Kompletny moduł apteki wraz z możliwością przeprowadzenia inwentaryzacji elektronicznej potrzebowałby okresu miesiąca na wykonanie. Oznacza to, że do końca roku 2015 możliwe byłoby ukończenie wstępnej, pełnej wersji części białej systemu informatycznego dla szpitala. Umożliwiło by to rozpoczęcie prac nad częścią szarą systemu informatycznego obsługi szpitala.

7. Opis implementacji i wdrożenia projektu

Baza danych wraz z aplikacją internetową obecnie działa na serwerze uruchomionym na układzie Raspberry Pi. Baza została stworzona przy użyciu MySQL Workbench i zarządzania zdalnie przy pomocy phpMyAdmin. Struktura bazy przedstawiona została w załączniku nr 1. Wykorzystywane są trzy typy głównych identyfikatorów: identyfikator pacjenta, numer księgi głównej i numer zdarzenia. Identyfikator pacjenta będący unikalny dla każdego pacjenta i pozwalający rozróżnić go w bazie danych. Numer księgi głównej identyfikujący konkretny pobyt pacjenta w szpitalu. Numer zdarzenia określający numer karty choroby, gdzie kart choroby może być wiele w ciągu jednego pobytu w szpitalu ze względu na pobyt na różnych oddziałach. Do określania rodzaju diagnozy i procedur medycznych wykorzystywane są uniwersalne kody ICD-10 i ICD-9. Aplikacja internetowa działa w oparciu o HTML, JavaScript oraz PHP. HTML i JavaScript wykorzystywane są do generowania stron WWW, przez które użytkownicy wchodzi w interakcje z systemem obsługi szpitala. Język PHP użyty został do zapewnienia komunikacji pomiędzy bazą, aplikacją internetową i mobilną. Mobilna aplikacja działa pod systemem iOS zainstalowanym w smartfonach Iphone. Napisana językiem Swift mając umożliwić jej szybsze działanie. Całość kodu implementacji można znaleźć na platformie GitHub pod adresem:

<https://github.com/ITApp0915SystemObslugiSzpitala/projekt>

System został stworzony w oparciu o trójwarstwowy model aplikacji. Gdzie serwer aplikacji i baza danych mają przydzieloną osobną warstwę, co ma za zadanie zwiększyć elastyczność i skalowalność stron WWW, przez które użytkownicy porusza się po systemie obsługi szpitala.



Rysunek 7.1 Trójwarstwowa architektura aplikacji do przetwarzania danych

Minimalne wymagania sprzętowe potrzebne dla systemu obsługi szpitala przedstawia tabela 7.1.

Tabela 7.1 Minimalne wymagania dla systemu obsługi szpitala

Procesor serwera	700 MHz
Pamięć operacyjna serwera	512 MB

Nośniki danych	Pojemność zależna ilości pacjentów i użytkowników jakich będzie obsługiwał system oraz typu plików przechowujących wyniki pacjenta
----------------	------------------------------------------------------------------------------------------------------------------------------------

Wielkość nośników przechowujących dane jest zależna od ilości użytkowników i pacjentów, o których to informacje będzie przechowywał system. Znaczenie ma również rodzaj plików przechowujących wyniki badań pacjenta generowane przez różnego typu urządzenia pomiarowe.

Serwer powinien być podłączony co najmniej do sieci lokalnej szpitala zapewniając pojedynczemu użytkownikowi transfer min. 10 Mb/sek.

W celu skorzystania z aplikacji internetowej niezbędny jest dowolny komputer będący w stanie obsłużyć dowolną przeglądarkę internetową na dowolnym systemie operacyjnym. Aplikacja mobilna obsługiwana jest zaś przez Iphony, urządzenia firmy Apple posiadające zainstalowany system iOS.

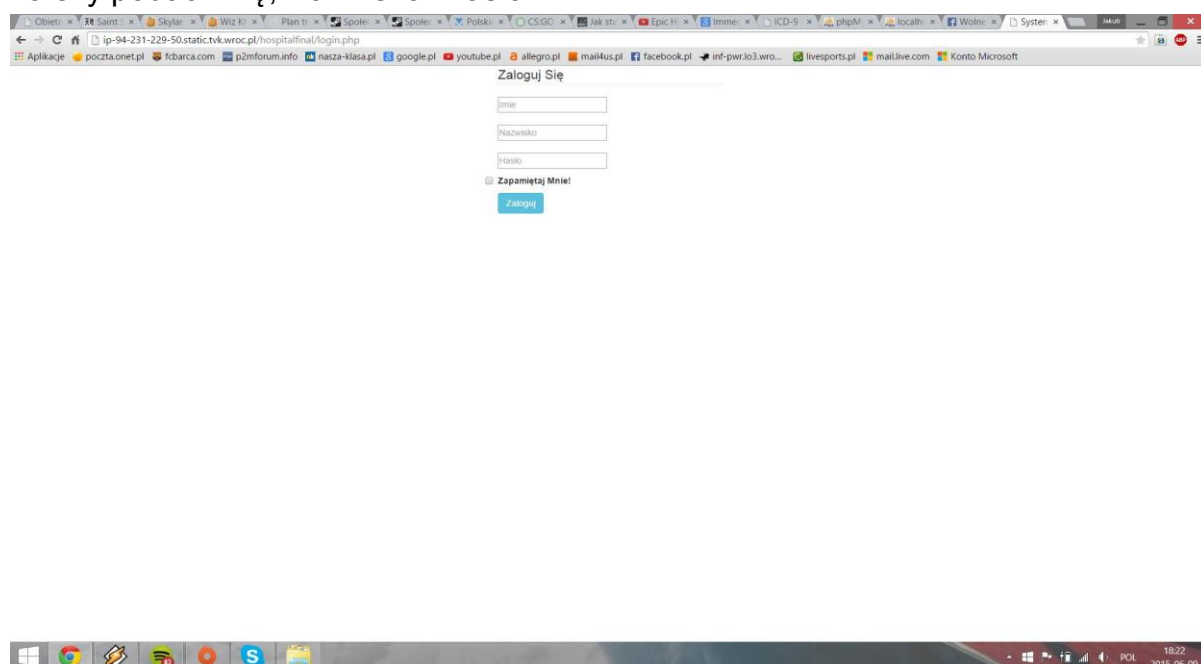
Tabela 7.2 Wymagania programowe systemu obsługi szpitala

Serwer	Linux OS, nginx
PC/laptop	dowolny OS, dowolna przeglądarka internetowa
Urządzenie mobilne	iOS

Uruchomienie systemu obsługi szpitala sprowadza się do uruchomienia przeglądarki internetowej na komputerze i połączeniu się z adresem:

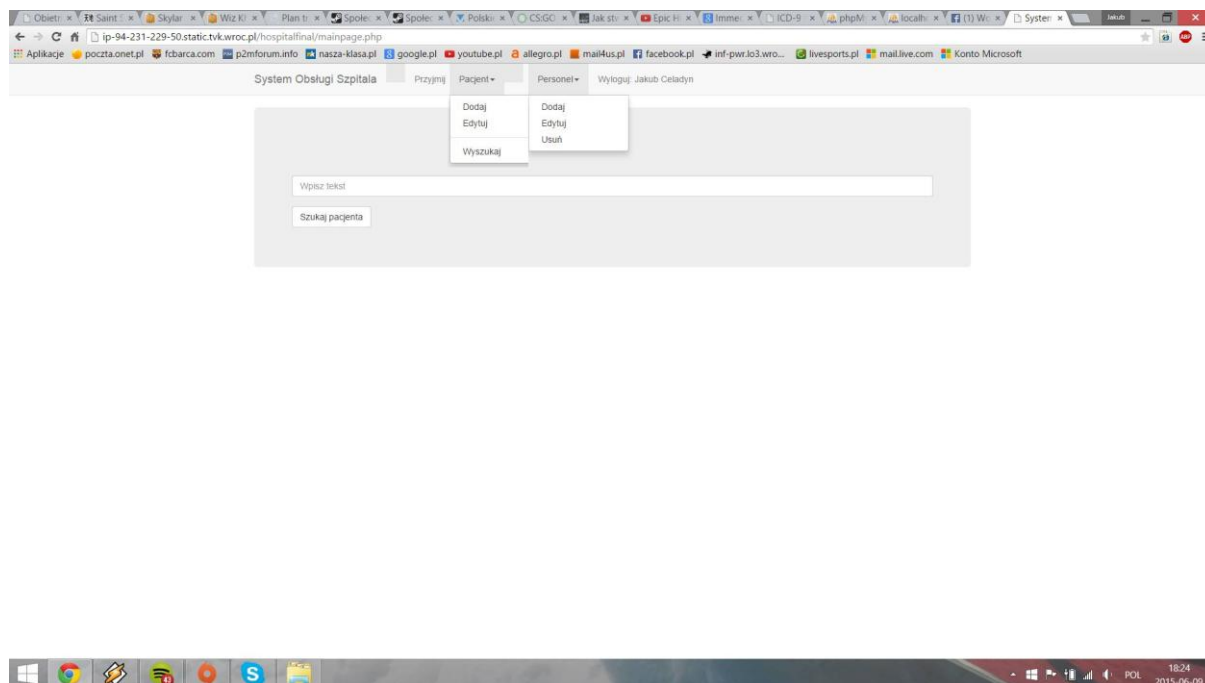
<http://ip-94-231-229-50.static.tvk.wroc.pl/hospitalfinal/login.php>

Po połączeniu się z adresem mamy ekran logowania, gdzie aby się zalogować należy podać Imię, Nazwisko i hasło.



Rysunek 7.2 Ekran logowania do aplikacji internetowej

Po poprawnym zalogowaniu przenosi nas na ekran początkowy aplikacji internetowej skąd można wykonywać wszystkie dostępne akcje dla konkretnego użytkownika.



Rysunek 7.2 Okno główne systemu obsługi szpitala

W przypadku aplikacji mobilnej należy na początku pobrać aplikację System obsługi szpitala i zainstalować przy pomocy AppStore. Polega to na wyszukaniu aplikacji w zbiorze i kliknięciu przycisku Instaluj. Gdy już aplikacja znajduje się na urządzeniu mobilny, wyszukujemy ją wśród zainstalowanych aplikacji i ją uruchamiamy poprzez kliknięcie na ikonkę aplikacji. W następnym kroku logujemy się do aplikacji mobilnej przy użyciu danych podanych przez administratora systemu. Po udanej próbie logowania możemy już korzystać ze wszystkich dostępnych nam funkcji aplikacji.

8. Wnioski

Praca nad tym projektem pozwoliła nam na udoskonalenie zdolności pracy grupowej. Ulepszyła sposób planowania etapów wykonywania poszczególnych elementów systemu i komunikacji pomiędzy różnymi grupami projektowymi wewnątrz zespołu. Lecz najważniejszą rzeczą jaką udało się nam wyciągnąć podczas prac jest wiedza na temat złożoności systemów informatycznych w gospodarce dla sektora publicznego jakim są szpitale.

Załącznik nr 1 Schemat bazy danych

