

POLITECHNIKA POZNAŃSKA
WYDZIAŁ INFORMATYKI
INSTYTUT INFORMATYKI



PRACA DYPLOMOWA INŻYNIERSKA

**MODUŁY DODATKOWE
DLA SYSTEMU ZARZĄDZANIA KARTĄ ELEKTRONICZNĄ
SMART CARD MANAGEMENT SYSTEM**

**Mateusz Kleinert
Szymon Kłosowski
Bartosz Łukaszewski
Paweł Winogrodzki**

Promotor:
prof. dr hab. inż. Joanna Józefowska

Poznań 2014

Spis treści

1	Wstęp	3
1.1	Wprowadzenie.	3
1.2	Opis problemu	4
1.3	Cel realizacji projektu	5
1.4	Przegląd istniejących rozwiązań.	6
1.5	Zakres pracy	7
1.6	Zespół realizujący projekt	7
1.7	Podział zadań w zespole programistów	7
2	Modelowanie biznesowe systemu	11
2.1	Aktorzy	11
2.2	Obiekty	11
2.3	Przypadki użycia	12
2.4	Problemy i koncepcja ich rozwiązania	12
2.5	Polityka prywatności w ujęciu biznesowym	15
3	Specyfikacja wymagań	17
3.1	Specyfikacja wymagań funkcjonalnych	17
3.1.1	Specyfikacja dotycząca SCMS	17
3.1.2	Specyfikacja dotycząca aplikacji klienckiej konferencji	18
3.1.3	Specyfikacja dotycząca aplikacji klienckiej Wielkopolskiego Centrum Onkologii.	19
3.2	Specyfikacja wymagań pozafunkcjonalnych	20
3.2.1	Kryteria i standardy jakości dla aplikacji klienckiej konferencji	21
3.2.2	Kryteria i standardy jakości dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii	22
3.2.3	Standardy kodowania	23
3.2.4	Wymagania dotyczące dokumentacji	24
4	Architektura systemu	27
4.1	SCMS	28
4.2	Aplikacja kliencka konferencji	28
4.3	Aplikacja kliencka Wielkopolskiego Centrum Onkologii	30

4.4	Komunikacja pomiędzy SCMS a aplikacjami klienckimi	31
4.5	Schemat bazy danych.	31
5	Opis implementacji	33
5.1	Narzędzia	33
5.2	Technologia.	33
5.3	Harmonogram projektu	34
5.4	Realizacja zadań.	34
5.5	Testy	35
5.6	Dokumentacja.	36
5.7	Podsumowanie realizacji.	37
6	Wnioski i zalecenia wdrożeniowe	39
6.1	Plan wdrożenia oprogramowania	39
6.2	Uwagi dotyczące użytkowania systemu	39
	Bibliografia	41
	Dodatek A Przypadki użycia	45
	Dodatek B Płyta CD	53

Wstęp

1.1 Wprowadzenie

Karty inteligentne, czyli karty z wbudowanym układem elektronicznym stosowane są w wielu dziedzinach życia już od kilkudziesięciu lat. Znajdujący się na karcie układ udostępnia szereg funkcji, począwszy od wspomagania operacji kryptograficznych po bezpieczne przechowywanie danych. Wraz z rozwojem nowych technologii coraz większą wagę przywiązuje się do kwestii bezpieczeństwa [6].

Bezpieczeństwo danych zyskuje coraz większy priorytet zwłaszcza w przedsiębiorstwach i dużych organizacjach. W tym miejscu możliwe staje się wykorzystanie kart inteligentnych jako bezpiecznego nośnika poufnych danych, tj. certyfikatów cyfrowych czy kluczy prywatnych użytkowników [17]. W zależności od potrzeb organizacji możemy odpowiednio personalizować wydawane karty, a tym samym nadawać oraz w późniejszym czasie nadzorować uprawnienia poszczególnych użytkowników.

Karty inteligentne najczęściej wykorzystuje się jako [6]:

- bankowe karty kredytowe,
- karty SIM w telefonach komórkowych,
- bilety w transporcie publicznym,
- karty identyfikacyjne,
- karty kontrolowanego dostępu.

Do ich głównych zadań możemy zaliczyć:

- bezpieczne uwierzytelnianie użytkownika lokalnego systemu,
- bezpieczne uwierzytelnianie zdalnego użytkownika, np. z wykorzystaniem wirtualnych sieci prywatnych,
- podpis i szyfrowanie poczty elektronicznej,
- zabezpieczanie transmisji typu klient-serwer,
- rejestrowanie akcji użytkownika, np. pomiar czasu pracy,
- kontrola dostępu do miejsc i sprzętu,
- bezpieczne przechowywanie poufnych danych.

Warto zwrócić uwagę na formy w jakich występują wspomniane karty. Obecnie mamy możliwość skorzystania z kart posiadających interfejs kontaktowy oraz bezkontaktowy (wykorzystujący fale radiowe).

Tak wszechstronne zastosowanie możliwe jest z uwagi na zapewnienie przez karty inteligentne szeregu usług. Jedną z najważniejszych i najbardziej użytecznych jest umożliwienie wykorzystania zasobów karty dopiero po podaniu kodu PIN. Niektóre karty posiadają również wbudowaną pamięć, której zawartość generowana jest na karcie, i której nie można w żaden sposób zmienić, ani skopiować na inne nośniki. Pozwala to na bezpieczne uwierzytelnianie użytkownika takiej karty. W połączeniu z metodami kryptograficznymi są one stosowane w bankach oraz innych instytucjach, które wymagają najwyższego poziomu bezpieczeństwa [5]. Istotną cechą kart inteligentnych jest także należyta dbałość o przechowywane na nich dane. Ważne jest nie tylko uchronienie ich przed nieuprawnionym odczytem, co realizuje się poprzez stosowne mechanizmy szyfrowania, ale także zapobieganie nieuprawnionej, czy też niezamierzonej modyfikacji. Karty takie zapewniają także ochronę danych w przypadku odłączenia karty od urządzenia aktualnie zapisującego na niej informacje. Uniknięcie tego zagrożenia realizowane jest za pomocą funkcji haszującej, którą porównuje się kompletność przesłanych danych [18][11].

Wszystko to sprawia, że zarządzanie kartami przez cały cykl ich życia może być skomplikowane. Aby usprawnić ten proces, projektuje się systemy do zarządzania kartami elektronicznymi, czyli stosowne oprogramowanie ułatwiające pracę z kartami oraz zawartymi na nich danymi.

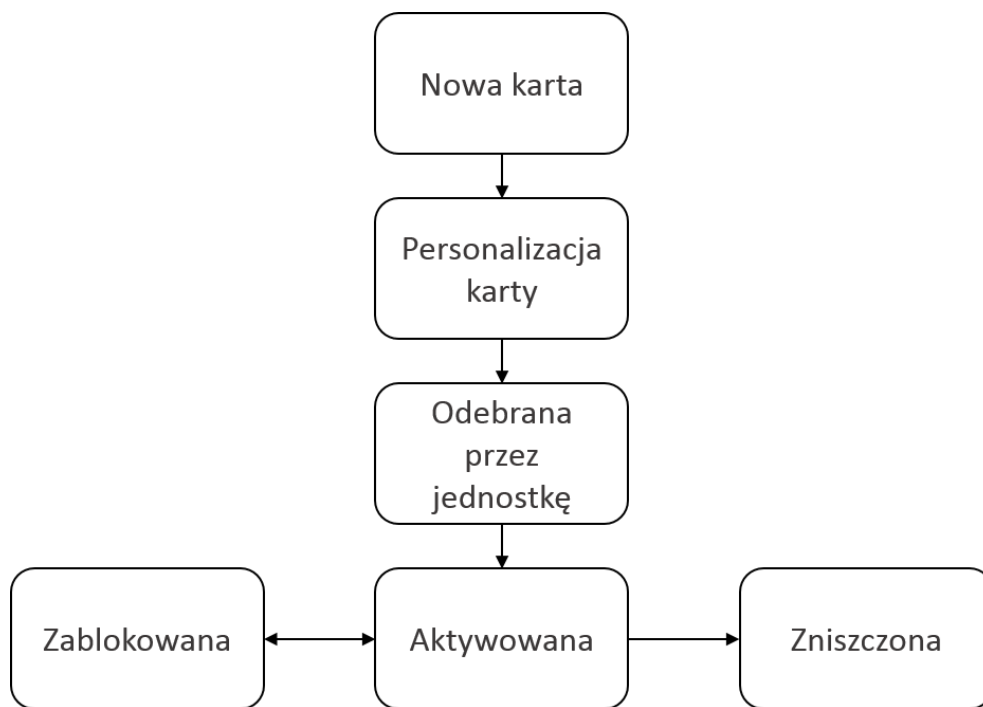
1.2 Opis problemu

Politechnika Poznańska posiada system - SCMS (Smart Card Management System) - służący do zarządzania kartami inteligentnymi, takimi jak karty pracownicze, studenckie oraz doktoranckie. Funkcjonalność systemu w obecnej wersji jest jednak ograniczona i umożliwia jedynie obsługę części wewnętrznych potrzeb uczelni. Wraz z coraz powszechniejszym stosowaniem kart inteligentnych pojawiła się propozycja rozbudowy systemu. Celem tej pracy inżynierskiej było stworzenie dodatkowego modułu umożliwiającego realizację zamówień na karty konferencyjne oraz rozbudowa systemu informatycznego Wielkopolskiego Centrum Onkologii w sposób umożliwiający automatyzację procesu zamawiania kart pracowniczych poprzez wykorzystanie uczelnianego systemu.

W procesie tworzenia karty producent ogranicza się do dostarczenia jej surowej wersji, która nie jest jeszcze przystosowana do spełnienia swojej właściwej funkcji. Aby móc korzystać z niej jako karty płatniczej lub identyfikacyjnej, wcześniej musi zostać ona poddana personalizacji. Proces ten polega na nadaniu karcie cech unikatowych, czyli profilu karty. Profil składa się z aplikacji, od których zależy funkcjonalność karty, informacji zapisanych we wbudowanym układzie elektronicznym oraz nadruku graficznego, będącego najczęściej również nośnikiem indywidualnych danych.

Stan karty elektronicznej może ulegać zmianie w czasie. Bardzo często już samo przeznaczenie karty determinuje zmiany stanu. Najlepszym przykładem są tutaj okresowo odnawiane bilety komunikacji publicznej. Należy również uwzględnić takie przypadki jak: zniszczenie blankietu, kradzież, zagubienie oraz inne sytuacje wymagające anulowania, czy

dezaktywacji karty. Obecny stan karty wraz z możliwymi przejściami z jednego stanu do drugiego określany jest jako cykl życia karty. Przykład takiego cyklu został przedstawiony na Rysunku 1.1.



Rysunek 1.1: Cykl życia karty elektronicznej

Każda karta zaczyna swój cykl życia od zebrania danych o jej profilu. Następnie czyste blankiety przechodzą przez proces personalizacji. Zlecenia personalizacji przesyłane są za pomocą programowych wtyczek bezpośrednio do drukarki. Wyposażone one są w specjalne czytniki, dzięki czemu rozpoznają model blankietu. Zewnętrzny program obsługujący czytnik umożliwia zapis danych w odpowiednich sektorach pamięci. Po zebraniu wszystkich potrzebnych informacji oraz danych następuje nadruk oraz instalacja aplikacji, czyli kompletne spersonalizowanie karty. Tak przygotowana karta jest gotowa do użycia przez użytkownika końcowego.

1.3 Cel realizacji projektu

Projekt posiada dwa główne cele realizowane współbieżnie przez członków zespołu projektowego. Pierwszym celem jest zaprojektowanie oraz wykonanie mechanizmu umożliwiającego prostą i intuicyjną obsługę zamówień związanych z kartami konferencyjnymi. W tym celu konieczne jest przygotowanie osobnej aplikacji klienckiej nastawionej na zamówienia większej liczby kart o podobnych właściwościach. W porównaniu do poprzedniej wersji systemu należy zmodyfikować sposób wprowadzania danych do aplikacji oraz zapewnić odpowiedni poziom bezpieczeństwa poprzez walidację danych pochodzących bezpośrednio od klienta. W celu zapewnienia wygody posługiwania się systemem przez użytkownika końcowego zakłada się zwiększenie dostępnej liczby źródeł danych dla zamawianych kart konferencyjnych.

Drugim celem projektu jest zintegrowanie istniejącego systemu zamawiania kart inteligentnych z aplikacją wewnętrzną Wielkopolskiego Centrum Onkologii oraz rozbudowanie go o nowe funkcjonalności wskazane przez klienta. Ta część projektu wymaga dostosowania istniejącego kodu aplikacji klienckiej do ograniczeń nałożonych przez istniejący już system, a także modyfikacji źródeł i sposobów pozyskiwania danych przez system. Założeniem projektu jest usprawnienie pracy użytkowników aplikacji klienckiej poprzez zautomatyzowanie jak największej części systemu. W celu spełnienia wymagań wskazanych przez klienta należy także wprowadzić nowy system ewidencji dla kart inteligentnych oraz zmodyfikować istniejący już system poprzez dodanie nowych funkcji. Wśród nich znajdują się między innymi: zmiana sposobu wyświetlania danych tabelarycznych, wprowadzenie szablonów wyszukiwania, zaimplementowanie funkcji służących do generowania raportów oraz list dla aktualnie wyświetlanych danych, czy też dodanie słownika statusów dla kart inteligentnych i pracowników.

Dodatkowo aplikacja kliencka Wielkopolskiego Centrum Onkologii zostanie rozszerzona o oprogramowanie służące do scentralizowanego zarządzania certyfikatami użytkowników. Odpowiedni moduł będzie miał za zadanie umożliwić zdalne wygenerowanie certyfikatu dla dowolnego użytkownika znajdującego się w domenie, pobranie tego certyfikatu oraz wgranie go na kartę użytkownika. Tak wygenerowany certyfikat będzie można odnowić oraz odwołać za pośrednictwem przygotowanego modułu. Dodatkowo wprowadzona zostanie osobna ewidencja certyfikatów oraz możliwość generowania raportów z nią związanych.

Obydwie części projektu są rozszerzeniem istniejącego systemu zarządzania kartami. Jednak Smart Card Management System w swojej pierwszej wersji przeznaczony był wyłącznie do realizowania wewnętrznych zleceń Politechniki Poznańskiej. Z tego względu jego struktura dopasowana została wyłącznie do wymagań stawianych przez Politechnikę Poznańską. Rozszerzenie funkcjonalności o obsługę klientów zewnętrznych wymaga zatem modyfikacji także po stronie aplikacji centralnej systemu. Zmianie należy poddać nie tylko interfejs, który dostosowany został do pracy z innym profilem użytkownika, ale także sposób obsługi zleceń.

Smart Card Management System w drugiej wersji projektowany jest z myślą o jego dalszym rozwoju. Celem projektu jest przygotowanie rozwiązania w pełni konfigurowalnego, a zatem umożliwiającego zaadaptowanie dla klientów o różnorodnych wymaganiach oraz realizowanie zleceń dla kart o różnym przeznaczeniu.

1.4 Przegląd istniejących rozwiązań

Z uwagi na coraz liczniejsze zastosowanie kart inteligentnych we wszystkich dziedzinach życia na rynku dostępnych jest wiele konkurencyjnych rozwiązań. Jednymi z bardzo zbliżonych rozwiązań są produkty firmy *CONTROL SYSTEM FMN*, która oferuje produkty dla legitymacji pracowniczych, studenckich, a także kart inteligentnych ogólnego zastosowania [9]. Systemy są dostępne w kilku wersjach w zależności od potrzeb użytkownika, jednak wszystkie oferują następujące funkcje:

- wydawanie nowych kart oraz ich duplikatów,
- personalizację elektroniczną i graficzną kart,

- zarządzanie cyklem życia karty,
- obsługę zewnętrznych baz danych, jak Excel, czy pliki CSV,
- obsługę standardu Barcode (kodów kreskowych),
- import plików graficznych z plików.

Są to oczywiście tylko niektóre z oferowanych funkcji, jednak w pełni pokrywają się one z funkcjami oferowanymi przez projekt realizowany w ramach tej pracy inżynierskiej. Na rynku dostępne są także inne systemy firm takich jak *OPTeam S.A*, czy *Global Payments Inc.*, lecz ich produkty ukierunkowane są często pod określonych odbiorców, jak uczelnie, czy banki. Należy również pamiętać, że każdy system wymaga indywidualnej instalacji oraz profesjonalnego wdrożenia zanim możliwe będzie korzystanie w pełni z jego usług.

1.5 Zakres pracy

Projekt zakłada zaprojektowanie dwóch aplikacji klienckich będących rozszerzeniem dla istniejących już systemów oraz przygotowanie dodatkowych rozszerzeń dla aplikacji wewnętrznej Wielkopolskiego Centrum Onkologii. Ważną kwestią jest zapewnienie uniwersalności przygotowywanego rozwiązania oraz możliwości jego późniejszej konfiguracji. Z punktu widzenia użytkownika końcowego projekt ma za zadanie automatyzację procesów związanych z zamawianiem, wydawaniem oraz późniejszym zarządzaniem kartami inteligentnymi. Podczas pracy należy zwrócić uwagę na specyficzne wymagania związane z różnymi typami obsługiwanych kart. Dodatkowo aplikacja kliencka Wielkopolskiego Centrum Onkologii wymaga integracji z już istniejącym oprogramowaniem.

1.6 Zespół realizujący projekt

Mateusz Kleinert - student kierunku Informatyka (I stopień), Wydział Informatyki, Politechnika Poznańska

Szymon Kłosowski - student kierunku Informatyka (I stopień), Wydział Informatyki, Politechnika Poznańska

Bartosz Łukaszewski - student kierunku Informatyka (I stopień), Wydział Informatyki, Politechnika Poznańska

mgr inż. Paweł Winogrodzki - student kierunku Informatyka (I stopień), Wydział Informatyki, Politechnika Poznańska

Autorzy pracy pragną serdecznie podziękować Działowi Rozwoju Oprogramowania Politechniki Poznańskiej oraz Działowi Informatyki Wielkopolskiego Centrum Onkologii za wsparcie merytoryczne oraz udostępnienie sprzętu niezbędnego do wykonania pracy.

1.7 Podział zadań w zespole programistów

Rozmiar i złożoność projektu wymagały odpowiedniego podziału pracy. Charakter poszczególnych części projektu pozwolił na możliwie duże zrównoleżenie pracy, oraz niez-

leżność w realizowaniu zadań przez członków zespołu. Realizacja projektu odbywała się zgodnie z następującym przydziałem zadań:

Mateusz Kleinert:

- implementacja protokołu komunikacyjnego,
- dodanie systemu logowania oraz sesji dla nowo utworzonego modułu aplikacji klienckiej,
- zmiana sposobu wyświetlania danych tabelarycznych poprzez umożliwienie zaawansowanego filtrowania oraz sortowania prezentowanych wyników,
- umożliwienie zapisywania parametrów wyszukiwania w postaci edytowalnych szablonów,
- dodanie mechanizmu umożliwiającego przygotowanie oraz złożenie zlecenia wydania kart elektronicznych, a także późniejsze zarządzanie tym zleceniem. Użytkownikowi aplikacji umożliwiono wykonywanie następujących czynności:
 - dodawanie do zlecenia kart serwisowych,
 - dodawanie do zlecenia kart pracowniczych,
 - edycja danych znajdujących się na kartach,
 - zlecenie wydania kart,
 - podgląd stanu zamówienia oraz stanu wchodzących w jego skład pojedynczych kart,
 - anulowanie zlecenia lub wybranych kart wchodzących w jego skład,
 - potwierdzenie odbioru zlecenia,
- automatyzacja procesu zamawiania oraz późniejszego ewidencjonowania zamówionych kart,
- zmiana sposobu ewidencji kart inteligentnych. W skład tego zagadnienia wchodzi:
 - zniesienie dotychczasowych ograniczeń dla kart serwisowych
 - wprowadzenie dodatkowych oznaczeń dla kart, np. typ karty, czy też status,
 - dodanie oraz umożliwienie zmiany statusu karty,
 - automatyzacja operacji powiązanych z danym typem karty,
- utworzenie systemu generowania raportów oraz zapisu wyników zapytania w formie dokumentu PDF oraz XLS,
- przygotowanie edytowalnego słownika statusów dla zleceń oraz kart inteligentnych,
- dodanie widoku prezentującego podgląd aktualnego stanu systemu,
- dodanie zintegrowanego widoku listy pracowników,
- zaprojektowanie mechanizmu umożliwiającego zarządzanie certyfikatami, w tym ich wydawanie i odnawianie,
- integracja systemu zarządzania certyfikatami z aplikacją wewnętrzną,
- przygotowanie dokumentacji dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii.

Szymon Kłosowski:

- logowanie do systemu,
- dodawanie użytkowników,
- wyświetlanie użytkowników,
- wczytywanie danych z pliku,
 - implementacja wczytywania danych z pliku,
 - testy dla wczytywania danych z pliku,
- wydruk dokumentu z zamówieniami w formacie PDF,
 - testy generowania dokumentu z zamówieniem,
 - implementacja generowania dokumentu zamówieniem,
- implementacja interfejsu tworzącego kartę o danym profilu,
- konfiguracja profilu aplikacji klienckiej kart konferencyjnych,
- przygotowanie dokumentacji dla aplikacji klienckiej kart konferencyjnych.

Bartosz Łukaszewski:

- historia zamówień,
 - sortowanie i grupowanie zamówień,
 - przeszukiwanie historii zamówień,
 - wyświetlanie historii zamówień,
- zlecenie kart do wydruku,
- zgłoszenie gotowych kart,
- wybór partii kart,
- wyświetlenie realizowanych zamówień,
- przygotowanie dokumentacji dla aplikacji centralnej.

Paweł Winogrodzki:

- podgląd historycznych zamówień,
- komunikacja z klientem,
- edycja kart,
 - implementacja mechanizmu edycji kart,
 - usuwanie kart,
 - testy dla edycji kart,
- wczytywanie danych z tekstu,
 - testy dla wczytywania danych z tekstu,
 - implementacja wczytywania danych z tekstu,
- generacja podglądu kart dla danej partii,

- przydzielenie profilu do aplikacji klienckiej kart konferencyjnych,
- dodawanie profilu aplikacji klienckiej kart konferencyjnych,
- dodawanie obsługi obrazów w formacie bmp,
- przygotowanie instrukcji instalacji dla aplikacji klienckiej kart konferencyjnych.

Modelowanie biznesowe systemu

2.1 Aktorzy

System w części dla kart konferencyjnych definiuje trzech aktorów:

- Administrator Międzyuczelnianego Centrum Personalizacji Legitymacji Studenckiej - osoba zarządzająca zarówno aplikacją kliencką, jak i aplikacją centralną,
- Użytkownik aplikacji klienckiej - osoba zalogowana i korzystająca z aplikacji klienckiej,
- Niezalogowany użytkownik aplikacji klienckiej - osoba niezalogowana do aplikacji.

W aplikacji klienckiej przeznaczonej dla Wielkopolskiego Centrum Onkologii zdefiniowanych jest jeden aktor:

- Administrator Wielkopolskiego Centrum Onkologii - osoba uprawniona do korzystania z aplikacji klienckiej Wielkopolskiego Centrum Onkologii.

2.2 Obiekty

Obiekty występujące w systemie różnią się zależnie od aplikacji klienckiej. Dla aplikacji klienckiej kart konferencyjnych zdefiniowane są następujące obiekty:

Profil karty - przypisany do karty i determinujący jej charakter zbiór aplikacji i wzorów graficznych. Aplikacje na karcie określają zestaw informacji potrzebnych do zdefiniowania zlecenia personalizacji karty, dane zapisane na karcie, jak i te udostępniane innym aplikacjom oraz sposób ich zapisu.

Wzór, wzór graficzny - wzór, który zostanie naniesiony na kartę i będzie wspólny dla wszystkich kart, korzystających z danego wzoru. Dodatkowo we wzorze definiowane są pola (np. nazwa konferencji, zdjęcie uczestnika), które zostają wypełnione różnymi danymi dla każdej personalizowanej karty. Wzory graficzne są przypisywane do profili kart i każdy profil może mieć przypisanych wiele wzorów.

Karta - spersonalizowana karta elektroniczna, nośnik danych w postaci plastikowej karty wraz z wbudowanymi układami scalonymi, nadrukiem oraz zapisanymi danymi.

Każda karta jest przypisana do jednego profilu karty i wykorzystuje jeden z wzorów graficznych dostępnych dla danego profilu.

Aplikacja - aplikacja na kartę jest nieodłączną częścią profilu. Zbiór wszystkich aplikacji i zależności między nimi wraz ze wzorami graficznymi tworzy profil karty, który determinuje jej charakter.

Zamówienie - zamówienie wydrukowania kart o określonym profilu, wzorze graficznym i zdefiniowanych wartościach pól dla każdej karty, jak również gotowe karty, które powstały na skutek wykonania tego zamówienia.

Dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii zdefiniowane są następujące obiekty:

Spersonalizowana karta - karta inteligentna wydana dla Wielkopolskiego Centrum Onkologii wraz z naniesionym wzorem graficznym oraz umieszczonymi na niej danymi.

Karta pracownicza - karta inteligentna powiązana z właściwym dla siebie pracownikiem Wielkopolskiego Centrum Onkologii. Karta ta zawiera dane pracownika oraz unikalny w obrębie systemu numer identyfikacyjny.

Karta serwisowa - karta inteligentna pełniąca rolę zapasowej karty pracowniczej. Karta ta nie posiada informacji o konkretnym pracowniku, a jedynie swój unikalny numer identyfikacyjny i jest wydawana tymczasowo w przypadku zniszczenia lub zgubienia właściwej dla danego pracownika karty inteligentnej.

Karta z certyfikatem - spersonalizowana karta z umieszczonym w pamięci certyfikatem cyfrowym zapewniająca kontrolę dostępu oraz ochronę umieszczonych na niej danych.

Zlecenie - zamówienie wydania karty lub grupy kart inteligentnych dla Wielkopolskiego Centrum Onkologii wraz z podaniem wszystkich danych koniecznych do przeprowadzenia procesu ich personalizacji. Karty inteligentne występujące w systemie możemy podzielić na karty pracownicze oraz serwisowe. Karty pracownicze dzielą się na karty pracowników wewnętrznych Wielkopolskiego Centrum Onkologii oraz karty pracowników firm zewnętrznych wykonujących prace w ramach umowy z Wielkopolskim Centrum Onkologii.

2.3 Przypadki użycia

Przypadki użycia można znaleźć w załączniku jakim jest *Dodatek A*.

2.4 Problemy i koncepcja ich rozwiązania

Podczas projektowania systemu dla Wielkopolskiego Centrum Onkologii zidentyfikowano punkty systemu, które mogą stanowić problem dla prawidłowego działania aplikacji.

Podstawowym problemem była kwestia komunikacji aplikacji klienckiej z aplikacją centralną. Po dokładniejszym zapoznaniu się z zagadnieniem okazało się, że istniejące już rozwiązanie wykorzystujące protokół AMQP oraz kolejki RabbitMQ jest wystarczające. W celu spełnienia wymagań klienta należało jedynie przepisać kod aplikacji komu-

nikacyjnej na kod odpowiadający standardom wymaganym przez Wielkopolskie Centrum Onkologii oraz zgodny z architekturą Zend Framework. Po dokonaniu tych zmian należało jeszcze odpowiednio skonfigurować aplikację kliencką oraz centralną.

Jednym z wymagań stawianych aplikacji był możliwie krótki czas potrzebny na wyświetlenie wyników dla widoków list pracowników, certyfikatów oraz kart inteligentnych. Dodatkowo listy te powinny umożliwiać sortowanie oraz filtrowanie prezentowanych danych. W celu rozwiązania tych problemów dokonano przeglądu istniejących rozwiązań umożliwiających prezentowanie danych w formie tabeli oraz późniejsze wykonywanie podstawowych operacji na zawartych w tabeli danych. Jako rozwiązanie wybrano plug-in DataTables [14] przeznaczony dla biblioteki jQuery. Jego duże możliwości konfiguracyjne pozwoliły na odpowiednie dostosowanie do wymagań stawianych przez klienta. W celu zapewnienia spójnego wyglądu aplikacji konieczna była modyfikacja kodu komponentu DataTables, tak aby generowane przez ten komponent znaczniki HTML (ang. HyperText Markup Language) posiadały odpowiednie atrybuty, a tekst zawarty między nimi był spójny. Modyfikacji poddano głównie funkcje odpowiedzialne za generowanie znaczników nawigacyjnych tabeli.

Komponent DataTables rozwiązuje kwestie związane z prezentowaniem danych oraz późniejszym sortowaniem i wyszukiwaniem odpowiednich wartości w tabeli. Nie rozwiązuje on jednak kwestii pobierania danych. Dane przechowywane są w bazie danych PostgreSQL w kilku tabelach połączonych ze sobą odpowiednimi relacjami. Do tej pory dane pobierane były z wykorzystaniem mechanizmów udostępnionych przez Zend Framework. Niestety takie podejście dla dużej ilości danych podzielonych na wiele tabel jest nieefektywne. W celu rozwiązania tego problemu przygotowano w bazie danych odpowiednie widoki, które pozwalają na pobranie tylko interesującej nas części danych. Dodatkowo dane te są odpowiednio formatowane przed przekazaniem ich do skryptu PHP. Takie podejście pozwoliło na kilkukrotne przyspieszenie procesu pobierania danych. W aplikacji zdecydowano się na pobieranie zawsze wszystkich danych. Takie podejście ma jedną wadę w postaci stosunkowo dużej ilości przesyłanych informacji. Warto jednak zwrócić uwagę na korzyści płynące z jednorazowego załadowania wszystkich danych. Do tej pory filtrowanie, czy też sortowanie danych wymagało przesłania odpowiedniego żądania do serwera i oczekiwania na odpowiedź. Decydując się na wykorzystanie komponentu DataTables pozbywamy się konieczności ciągłego wysyłania żądań. Wszystkie operacje wykonywane są na komputerze klienta, co w tym wypadku znacznie przyspiesza pracę systemu. Warto pamiętać także o tym, że widoki prezentujące listy pracowników, certyfikatów, czy też kart inteligentnych nie są często odświeżane, a zatem konieczność załadowania wszystkich danych występuje stosunkowo rzadko. Filtrowanie prezentowanych użytkownikowi wyników odbywa się natomiast wielokrotnie.

Kolejnym problemem była kwestia połączenia oprogramowania służącego do zarządzania certyfikatami użytkowników z aplikacją Wielkopolskiego Centrum Onkologii. Należało zapewnić komunikację aplikacji klienckiej z odpowiednim urzędem certyfikacji oraz zaprojektować interfejs, który pozwoli wykonywać operacje związane z zarządzaniem certyfikatami. Wymaganiem stawianym przez klienta było przygotowanie rozwiązania, które pozwoli z poziomu aplikacji klienckiej na: wydanie, odnowienie, pobranie oraz odwołanie certyfikatu dowolnego użytkownika. W tym przypadku zdecydowano się wykorzystać technologię .NET i zaimplementować aplikację typu Web Service [20]. Dla tego rozwią-

zania kluczową kwestią było nadanie odpowiednich uprawnień aplikacji oraz utworzenie użytkownika, który ma prawo wystawiać certyfikaty dowolnemu innemu użytkownikowi posiadającemu konto w usłudze AD (ang. Active Directory). Ważną kwestią było także zapewnienie bezpieczeństwa takiej usługi. Z tego względu dostęp do usługi został ograniczony tylko do wskazanej puli adresów IP.

Podczas prac nad aplikacją typu Web Service napotkano na problem, który nie został rozwiązany. Z poziomu tej aplikacji użytkownik nie jest w stanie odwołać certyfikatu. Problem ten próbowano rozwiązać poprzez zastosowanie odpowiednich bibliotek DLL (ang. Dynamic-Link Library) jednak standardowy zestaw bibliotek nie zawiera takiej funkcji. Przeszukanie informacji zawartych w dokumentacji firmy Microsoft także nie przyniosło pozytywnych rezultatów. Problem próbowano rozwiązać także poprzez wykorzystanie wbudowanego w system Windows Server 2012 narzędzia o nazwie certutil [10]. Narzędzie to pozwala na zaawansowane zarządzanie certyfikatami systemowymi oraz użytkownika. W celu zastosowania tego rozwiązania konieczne było wykonanie polecenia systemowego z poziomu kodu aplikacji typu Web Service. Taka akcja okazała się niemożliwa do wykonania ze względu na zabezpieczenia systemu. Nawet po przyznaniu zwiększonych praw dostępu dla aplikacji, oraz wyłączeniu części zabezpieczeń systemowych wykonanie tego polecenia nie było możliwe. Po omówieniu zagadnienia z klientem oraz kierownikiem projektu zdecydowano o zrezygnowaniu z tej funkcji systemu.

Jednym z wymagań stawianych aplikacji była możliwość generowania raportów w formie plików PDF oraz XLS. W tym celu wykorzystano biblioteki PHPExcel [15] oraz TCPDF [16].

W przypadku aplikacji klienckiej kart konferencyjnych problemy związane z komunikacją między serwerem, a klientem, wykorzystywanym systemem zarządzania bazą danych, jak również prezentacją informacji przechowywanych w bazie danych zostały rozwiązane w taki sam sposób, jak w podstawowej wersji systemu zarządzania kartami elektronicznymi [1]. Stało się tak z uwagi na wykorzystanie tego samego frameworku PHPLiteMVC i podobieństwo wymagań między już istniejącym modulem zarządzania kartami pracowniczymi Politechniki Poznańskiej, a modulem zarządzania kartami konferencyjnymi.

Moduł komunikacyjny przesyłający informacje między aplikacją centralną, a klientem kart konferencyjnych wymagał jedynie rozbudowy o możliwość blokowania wybranych użytkowników klienta i przypisywanie każdemu użytkownikowi dowolnego podzbioru profili, z których może korzystać, a które są powiązane z nowym modulem. Funkcja ta nie była dotychczas obsługiwana przez pierwotny system. Dodatkowo należało również stworzyć nową kolejkę RabbitMQ dla utworzonego modułu i osobny klucz szyfrujący, umożliwiający bezpieczne przesyłanie danych.

Jako system zarządzania bazą danych został ponownie wykorzystany PostgreSQL. Podobnie kwestia prezentacji danych po stronie klienta została rozwiązana przez wykorzystanie komponentu DataTables uzupełnionego o własnoręcznie dodane akcje w języku jQuery, które były niezbędne w celu zwiększenia ergonomii aplikacji i intuicyjności obsługi.

Głównym problemem w module kart konferencyjnych było dodanie możliwości wczytywania danych, które mają się pojawić na karcie, z nowych źródeł: plików w formatach CSV, XLS i XLSX. Dodatkowo system musiał umożliwiać wczytywanie obrazów z archiwów w formacie ZIP, jak również ręczne dodawanie i ewentualną edycję wprowadzonych

danych. W celu obsługi wczytywania danych tekstowych z plików, zapisanych w wyżej wymienionych formatach, zostały wykorzystane: biblioteka PHPExcels w wersji 2.1 i własna implementacja obsługi plików w formacie CSV. Ostateczne rozwiązanie wymaga podania wszystkich danych tekstowych w pliku, zaś w przypadku obrazów należy zapisać nazwy plików, które muszą się znajdować w osobno wczytywanych archiwach. Ręczna edycja wprowadzonych danych wymagała dopisania funkcji w języku jQuery, rozszerzających możliwości wtyczki DataTables, gdyż podstawowa, darmowa wersja tego komponentu nie posiadała takich możliwości [14].

2.5 Polityka prywatności w ujęciu biznesowym

Z uwagi na operowanie systemem na danych osobowych, zarówno uczestników konferencji, jak również pracowników Wielkopolskiego Centrum Onkologii, bardzo istotną rolę w projekcie jest zapewnienie odpowiedniego poziomu bezpieczeństwa. Podstawą prawną do tej kwestii były:

- Ustawa z dnia 29 sierpnia 1997 r. o ochronie danych osobowych (Dz. U. 1997 nr 133 poz. 883 z późniejszymi zmianami),
- Rozporządzenie Ministra Spraw Wewnętrznych i Administracji z dnia 29 kwietnia 2004 r. w sprawie dokumentacji przetwarzania danych osobowych oraz warunków technicznych i organizacyjnych, jakim powinny odpowiadać urządzenia i systemy informatyczne służące do przetwarzania danych osobowych (Dz. U. 2004 nr 100 poz. 1024).

Moduły aplikacji komunikują się ze sobą za pomocą aplikacji komunikacyjnej, gwarantującej niezawodność dostarczania wiadomości. Dodatkowo wiadomości są szyfrowane symetrycznym szyfrem blokowym AES (ang. Advanced Encryption Standard). Klucze potrzebne do szyfracji i deszyfracji wiadomości przechowywane są w lokalnej bazie danych w postaci zahaszowanej. Dostęp do nich mają wyłącznie upoważnione osoby.

W celu komunikacji aplikacji internetowych z przeglądarką użytkownika zastosowany jest protokół HTTPS, będący szyfrowaną wersją protokołu HTTP. Gwarantuje to bezpieczeństwo podczas przekazywania danych wprowadzanych przez użytkownika.

Po stronie aplikacji centralnej, oraz aplikacji klienckich przechowywanie danych następuje za pomocą bazy danych *PostgreSQL*, znajdującej się w sieci lokalnej każdego z modułów. Całość systemu wyposażona jest także w zestaw zapór ogniowych, uniemożliwiający dostęp do danych przez osoby nieupoważnione. Dane szczególnie newralgiczne, jak hasła użytkowników przechowywane są w postaci zaszyfrowanej.

W aplikacji klienckiej Wielkopolskiego Centrum Onkologii występuje dodatkowo konieczność ograniczenia dostępu do usługi umożliwiającej zdalne zarządzanie certyfikatami użytkowników. Usługa wykonana jest w technologii Web Service, a dostęp do niej ograniczony jest do konkretnej puli adresów IP (ang. Internet Protocol) zdefiniowanej przez administratora systemu po stronie Wielkopolskiego Centrum Onkologii.

Specyfikacja wymagań

3.1 Specyfikacja wymagań funkcjonalnych

We wstępnej fazie projektu zespół menedżerski dokonał analizy powstającego systemu pod względem wymagań funkcyjnych. Przeprowadzanie takiej analizy umożliwiło określenie zbioru funkcji, które nowe moduły powinny implementować. Rozpisanie wymagań funkcjonalnych projektu umożliwiło na określenie sposobu interakcji użytkownika z nowymi elementami systemu, szczegółowe określenie wszystkich dostępnych użytkownikowi akcji i podział zadań między członków grupy inżynierskiej. Dodatkowo zdefiniowanie tych wymagań pozwoliło na łatwiejszą analizę ewentualnych braków systemu, uszczegółowienie kwestii bezpieczeństwa i wychwycenie potencjalnych błędów we wczesnych fazach projektu, dzięki czemu możliwe jest uniknięcie poważniejszych błędów w części implementacyjnej pracy.

Wymagania funkcjonalne zostały podzielone ze względu na moduł i aktora, który wykorzystuje daną funkcję systemu. Każde wymaganie zawiera definicję jednej konkretnej akcji, która miała być możliwa do wykonania w nowych elementach systemu, wraz z ewentualnymi alternatywnymi scenariuszami. Ponadto zostały określone sytuacje, do których wystąpienia system nie miał prawa dopuszczać.

3.1.1 Specyfikacja dotycząca SCMS

Wymagania względem centralnej części system SCMS nie uległy poważnym zmianom względem swojej poprzedniej wersji. Zadaniem tej części systemu jest nadal pozwalać administratorowi na zarządzanie całym systemem i gromadzenie zleceń z wszystkich modułów klienckich (zarówno już istniejących, jak i tych, które mogą powstać w przyszłości). W celu zachowania wyższego poziomu bezpieczeństwa administrator systemu nadal może logować się jedynie z sieci lokalnej po podaniu nazwy i hasła. Po pomyślnym uwierzytelnieniu system umożliwia dostęp do funkcji podzielonych na moduły opisane poniżej.

Moduł administracyjny

Ten moduł umożliwia między innymi zarządzanie aplikacjami klienckimi, z którymi powiązana jest centralna część systemu. W tej części można zdefiniować np. nazwy kolejek RabbitMQ i klucze szyfrujące wymagane do bezpiecznego przesyłania danych między aplikacją centralną, a kliencką. Możliwe jest również definiowanie wymagań nakładanych

na hasła użytkowników wszystkich aplikacji klienckich bądź określenie, czy zlecenia od modułów klienckich wymagają uwierzytelnienia użytkownika zanim zostaną przyjęte.

Nową opcją dodaną podczas prac nad obecną wersją systemu jest możliwość zarządzania użytkownikami aplikacji klienckich. Została dodana możliwość zdefiniowania nowych użytkowników, ustawiania i zmiany ich haseł, jak również blokowania dostępu użytkownikom do aplikacji klienckiej. Dodatkowo każdy użytkownik może mieć teraz określony dostęp do dowolnego podzbioru profili kart, przypisanych do danego modułu klienckiego, z którego korzysta użytkownik.

Konfiguracja modułu drukującego

Ten element umożliwia konfigurację i zarządzanie stanowiskami drukującymi, które drukują partie kart wysłanych do nich przez moduł centralny. Możliwe jest dodawanie, usuwanie i edycja stanowisk drukujących.

Moduł konfiguracji profili kart

Ta część panelu administracyjnego umożliwia zarządzanie profilami kart, które są następnie udostępniane do wykorzystania przez użytkowników aplikacji klienckich. Opcje dostępne w tym module pozwalają ponadto na tworzenie i przypisywanie wzorów graficznych do profili, definiowanie blankietów kart oraz dodawanie aplikacji, które mają zostać wgrane na karty w trakcie personalizacji kart utworzonych dla danego profilu.

Moduł zarządzania zleceniami

W tym module administrator systemu może przeprowadzać wszystkie operacje dostępne dla kart od momentu złożenia zamówienia do momentu wysłania gotowych kart do klienta. Ten fragment systemu pozwala na podział zamówień na partie kart, przydział partii do wybranych stacji drukujących, a następnie na przyjęcie bądź odrzucenie wydrukowanej partii i odesłanie gotowych kart do klienta, który złożył na nie zamówienie. W obecnej wersji ten moduł został poddany modyfikacjom w celu zwiększenia ergonometrii i przejrzystości systemu. Liczba funkcji została taka sama, lecz zmniejszono liczbę ekranów, z których administrator musi korzystać. Dodana została również historia zrealizowanych zamówień w celu lepszego śledzenia działania systemu [1].

3.1.2 Specyfikacja dotycząca aplikacji klienckiej konferencji

Pierwszym z dwóch nowych modułów klienckich dodanych podczas wykonywania tego projektu był moduł kliencki kart konferencyjnych. Wymagania funkcjonalne stawiane przed tym elementem systemu były po części zbliżone do wymagań funkcyjnych klienta kart pracowniczych, lecz pojawiły się również dodatkowe wymagania i niektóre uległy modyfikacjom. W celu zalogowania się użytkownik może nadal korzystać z systemu eLogin bądź posiadać konto zdefiniowane w centralnym module systemu SCMS. Aplikacja kliencka kart konferencyjnych została podzielona na dwa główne moduły.

Moduł zarządzania kartami

Ten moduł umożliwia zalogowanemu użytkownikowi na tworzenie nowych zamówień i śledzenie przebiegu personalizacji kart na podstawie informacji otrzymanych od aplikacji centralnej. Głównym nowym wymaganiem w przypadku tego elementu było umożliwienie użytkownikowi wczytywania dowolnych danych na karty niezależnie od profilu i wzoru graficznego, które je definiowały. Dodatkowo użytkownik miał mieć możliwość wczytywania danych z plików, bądź wprowadzania ich ręcznie. Ponadto system miał pozwalać na

podgląd i dowolną edycję danych przed stworzeniem zamówienia i ostatecznego wysłania go do modułu centralnego w celu utworzenia kart. Po zatwierdzeniu zamówienia wymaganą funkcją tej części aplikacji było śledzenie stanu całego zamówienia, jak i każdej karty z osobna. Po otrzymaniu wydrukowanych kart użytkownik dostał również możliwość potwierdzenia ich odbioru. Następnie użytkownik mógł zarządzać potwierdzonymi kartami, czyli np. aktywować je, niszczyć lub tworzyć duplikaty.

Moduł administracyjny

Poza ustawianiem kluczy szyfrujących i kolejek aplikacji RabbitMQ ten moduł miał za zadanie umożliwiać synchronizację profili przypisanych danemu użytkownikowi. Z uwagi na fakt, że użytkownik aplikacji klienckiej może mieć dostęp jedynie do podzbioru wszystkich profili kart przypisanych do klienta, wymagane było, aby uniemożliwić mu synchronizację i zdobycie wiedzy o innych profilach, do których nie powinien mieć dostępu.

3.1.3 Specyfikacja dotycząca aplikacji klienckiej Wielkopolskiego Centrum Onkologii

Aplikacja kliencka Wielkopolskiego Centrum Onkologii została podzielona na dwie główne części, dla których przygotowano odrębne zestawy wymagań. Głównym wymaganiem stawianym aplikacji było zautomatyzowanie procesu zamawiania kart pracowniczych.

Z tego względu pierwszą część całej aplikacji stanowił moduł komunikacyjny. Jego zadaniem było umożliwienie przesyłania komunikatów pomiędzy aplikacjami kliencką i centralną. Kolejnym krokiem było dostosowanie modułu wykorzystywanego do zamawiania kart do aktualnych potrzeb organizacji. W tym celu należało zapoznać się z przebiegiem procesu zamawiania kart pracowniczych przez klienta, wyznaczyć momenty tego procesu, w których klient chce otrzymywać informacje od systemu oraz ostatecznie zidentyfikować informacje, które system ma prezentować. Pierwszym z wymagań dotyczących procesu zamawiania kart pracowniczych było umożliwienie zamawiania pojedynczej karty oraz grupy kart. Użytkownik powinien mieć możliwość wybrania kart, które chce zamówić, a następnie dodania ich do zlecenia. Do momentu zatwierdzenia zlecenia, czyli przesłania go do aplikacji centralnej, użytkownik może dowolnie dodawać nowe karty do zlecenia, usuwać karty wcześniej dodane do zlecenia oraz edytować dane powiązane z kartami znajdującymi się w zleceniu. Wcześniejsza wersja aplikacji centralnej pozwalała jedynie na zamawianie pojedynczych kart, tzn. zlecenie wydania grupy kart było traktowane przez aplikację centralną jako wiele zleceń wydania pojedynczej karty. Po konsultacjach z pracownikami Działu Informatyki Wielkopolskiego Centrum Onkologii zdecydowano się na zmianę tego rozwiązania.

Kolejnym wymaganiem związanym z procesem zamawiania kart było dostarczenie mechanizmu sprawdzania danych umieszczanych na kartach w momencie dodawania ich do zlecenia. Jest to wymaganie, które zostało dodane w czasie prac nad projektem i nie ogranicza się tylko do sprawdzenia, czy przesyłane dane są poprawne z punktu widzenia aplikacji centralnej. To wymaganie ma na celu uniemożliwić zlecenie wydania nowej karty w przypadku, gdy w systemie znajdują się już aktywne karty dla danego pracownika lub w przypadku wysłania wcześniej zlecenia wydania karty dla danego pracownika.

Karty pracownicze posiadają obowiązkowe pole ze zdjęciem pracownika. Aplikacja Wielkopolskiego Centrum Onkologii nie przechowywała zdjęć pracowników. Chcąc moż-

liwie maksymalnie zautomatyzować proces wydawania kart pracowniczych, do aplikacji dodano możliwość zarządzania zdjęciami pracowników.

Bardzo ważną kwestią z punktu widzenia klienta była możliwość podglądu karty przed wysłaniem zlecenia. Miało to stanowić miejsce, w którym użytkownik ostatecznie potwierdza poprawność danych umieszczonych na karcie.

Po wysłaniu zlecenia jego status jest monitorowany. Użytkownikowi aplikacji klienckiej umożliwiono sprawdzenie statusu zamówienia oraz statusów konkretnych kart zawartych w zamówieniu. Możliwe statusy zostały zaproponowane przez klienta, jednak uległy one zmianie w czasie prac nad projektem. Było to konsekwencją konieczności dostosowania się do zmian zachodzących w aplikacji centralnej oraz zmian zasugerowanych przez klienta.

Statusy stanowią odrębne wymaganie dla aplikacji. W module kart inteligentnych występują statusy dla zamówień oraz samych kart inteligentnych. Należało umożliwić użytkownikowi końcowemu kontrolę nad tymi statusami poprzez wprowadzenie konfigurowalnego słownika.

Wymagania określają także sposób zmiany ewidencji kart inteligentnych. W nowym systemie został wprowadzony odmienny mechanizm zarządzania kartami w zależności od ich typu. W ten sposób zniesiono wcześniejsze ograniczenia nałożone na karty serwisowe oraz ułatwiono zarządzanie nimi poprzez możliwość przypisania ich do dowolnego pracownika lub pozostawienia nieprzypisanymi.

Ostatnim wymaganiem stawianym modułowi kart inteligentnych jest możliwość tworzenia raportów. W widoku pracowników oraz kart inteligentnych możemy dowolnie filtrować prezentowane dane, a następnie zapisać wartości filtrów w formie szablonu lub bezpośrednio z poziomu danego widoku wygenerować dokument z raportem. Raporty można generować do dwóch formatów, tj. PDF oraz XLS.

Drugą część aplikacji stanowił moduł certyfikatów. W tej części wymagane było udostępnienie mechanizmu wydawania certyfikatów użytkownikom systemu, ich wznawianie oraz prowadzenie dodatkowej ewidencji certyfikatów ze szczególnym uwzględnieniem ich daty ważności. Należało także zapewnić mechanizm powiadomień o wygasających certyfikatach.

W czasie trwania prac dodano także wymaganie dotyczące systemu logowania. Początkowo nie brano tego pod uwagę, a dostęp do modułu kart inteligentnych miał być udzielony każdemu użytkownikowi głównej aplikacji Wielkopolskiego Centrum Onkologii. Z czasem zdecydowano jednak ograniczyć ten dostęp poprzez wprowadzenie mechanizmu podwójnego logowania.

3.2 Specyfikacja wymagań pozafunkcyjnych

Oprócz wymagań funkcjonalnych duży wpływ na ostateczny produkt mają wymagania pozafunkcyjne, czyli niezwiązane bezpośrednio z technicznymi aspektami aplikacji. Dotyczą one głównie interfejsu użytkownika, jego charakterystyki i wyglądu. Określają interfejsy sprzętowe i programowe użyte do tworzenia systemu oraz regulują maksymalny dopuszczalny czas odpowiedzi systemu. Kolejnym aspektem wymagań pozafunkcyjnych jest określenie poziomu niezawodności projektowanego systemu. Wymagania dotyczące bezpieczeństwa projektowanego systemu, stopień ochrony danych i poufności ich

przetwarzania również ujęte są w wymaganiach pozafunkcyjnych. Właściwie zdefiniowane i ściśle przestrzegane podczas implementacji wymagania pozafunkcyjne są bardzo istotne dla całego projektu.

3.2.1 Kryteria i standardy jakości dla aplikacji klienckiej konferencji

Aplikacja kart konferencyjnych posiada zdefiniowane następujące standardy jakości:

Interfejsy użytkownika

Wymagane jest, aby aplikacja kliencka posiadała interfejs webowy (cienki klient) - w formie stron www. Graficzny interfejs użytkownika musi być spójny z dotychczasowymi rozwiązaniami oprogramowania Politechniki Poznańskiej (np. eLogin, eStudent 2.0). Szczegółowe wytyczne opisuje standard: SIW/CI PP. Strony www będą przygotowane w sposób wspierający wielojęzyczność. Przygotowane będą treści w języku polskim. Dla formularzy w których pojawia się lista elementów, na których wykonywane są operacje, które mogą być wykonywane grupowo - powinna istnieć możliwość grupowego wykonania tych operacji. Nawet jeżeli dopuszczono odstępstwo w tym zakresie to sposób wywołania funkcji wykonującej daną operację powinien umożliwiać jej wywołanie dla wielu wierszy listy. Dla widoku list powinna istnieć wprost możliwość sortowania list po kolumnach wyświetlanych na liście. Dla widoku list powinna być możliwość wyszukiwania po zaawansowanych kryteriach - kryteria powinny być możliwe do ustalenia przez administratora systemu lub użytkownika. Dla widoku list powinna istnieć możliwość eksportu zawartości do plików edytowalnych (np. xls, csv).

Interfejsy sprzętowe

Drukowanie kart będzie się odbywało poprzez moduł SCMS printer daemon.

Interfejsy programistyczne

Aplikacja kliencka nie będzie wspierała zewnętrznych interfejsów programistycznych.

Interfejsy komunikacyjne

Specyfikacja protokołu komunikacyjnego, zdefiniowana przez Dział Rozwoju Oprogramowania Politechniki Poznańskiej, znajduje się w dokumencie dostępnym na stronie projektu [4].

Wydajność

Należy zapewnić odpowiedni poziom szybkości działania aplikacji. Przy założeniu pięciu równocześnie aktywnych użytkowników na standardowej maszynie serwerowej (Xeon E3500, 8GB RAM, Linux, dwa łącza 100Mbps) czas odpowiedzi powinien wynieść nie więcej niż 10 sekund w 90% przypadków. Jeżeli czas odpowiedzi musi zostać przekroczony, należy powiadomić użytkownika o spodziewanym wydłużonym czasie obsługi żądania. Nie jest dopuszczalna realizacja żądania w ramach sesji http w czasie dłuższym niż 1 minuta. Krytyczne operacje powinny być dodatkowo potwierdzane przez użytkownika.

Zgodność

Wymagana jest kompatybilność z przeglądarką internetową Firefox.

Bezpieczeństwo

Wymagane jest, aby system spełniał wymagania ustawy o ochronie danych osobowych.

Poufność

Komunikacja z systemem będzie realizowana za pomocą szyfrowanej wersji protokołu HTTP - HTTPS.

Inne wymagania

Dla aplikacji zdefiniowano dwa dodatkowe wymagania dotyczące wykorzystywanej terminologii oraz standardu kodowania:

- wymagane jest zachowanie terminologii zgodnej z podaną na stronie projektu [19],
- wymagane jest przyjęcie standardu kodowania podanego na stronie projektu [12]. Powyższe standardy również zostały zdefiniowane przez Dział Rozwoju Oprogramowania.

3.2.2 Kryteria i standardy jakości dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii

Dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii zdefiniowane kryteria i standardy jakości znajdują się poniżej:

Interfejsy użytkownika

Wymagane jest, aby aplikacja kliencka posiadała interfejs webowy (cienki klient) - w formie stron WWW (ang. World Wide Web). Graficzny interfejs użytkownika musi być spójny z dotychczasowymi rozwiązaniami stosowanymi w głównej aplikacji Wielkopolskiego Centrum Onkologii. Strony WWW nie będą wspierały wielojęzyczności. Przygotowane będą treści w języku polskim. Dla formularzy w których pojawia się lista elementów, na których wykonywane są operacje, które mogą być wykonywane grupowo - powinna istnieć możliwość grupowego wykonania tych operacji. Nawet jeżeli dopuszczono odstępstwo w tym zakresie to sposób wywołania funkcji wykonującej daną operację powinien umożliwiać jej wywołanie dla wielu wierszy listy. Dla widoku list powinna istnieć wprost możliwość sortowania list po kolumnach wyświetlanych na liście. Dodatkowo dla widoku list powinna być możliwość wyszukiwania po zaawansowanych kryteriach - kryteria powinny być możliwe do ustalenia przez Administratora Wielkopolskiego Centrum Onkologii.

Interfejsy sprzętowe

Wymagane jest, aby produkt współpracował z urządzeniem lub oprogramowaniem służącym do wydawania certyfikatów stosowanym przez Wielkopolskie Centrum Onkologii. Przez oprogramowanie rozumiemy system operacyjny Windows Serwer 2012 oraz wchodzące w jego skład urządzenie certyfikacji CA (ang. Certificate Authority) i usługę katalogową AD (ang. Active Directory).

Interfejsy programistyczne

Aplikacja kliencka musi utrzymać zgodność z obecnie istniejącymi interfejsami programistycznymi, np. interfejsem pobierającym dane pracowników. Potrzebne interfejsy są już zaprojektowane w istniejącej aplikacji. Jedynym wyjątkiem jest interfejs do komunikacji z oprogramowaniem służącym do wydawania certyfikatów.

Interfejsy komunikacyjne

Sposób komunikacji między aplikacją kliencką a serwerem, a także wymagane protokoły komunikacyjne przedstawiono na diagramie architektury systemu. Specyfikacja protokołu komunikacyjnego znajduje się w dokumencie dostępnym na stronie projektu [4].

Wydajność

Krytyczne operacje powinny być dodatkowo potwierdzane przez użytkownika.

Zgodność

Wymagana jest kompatybilność z następującymi przeglądarkami internetowymi:

- Google Chrome,
- Firefox,
- Internet Explorer,
- Opera.

Bezpieczeństwo

Wymagane jest, aby system spełniał wymagania ustawy o ochronie danych osobowych. Krytyczne operacje powinny być dodatkowo potwierdzane przez użytkownika.

Inne wymagania

Dla aplikacji zdefiniowano dwa dodatkowe wymagania dotyczące wykorzystywanej terminologii oraz standardu kodowania:

- wymagane jest zachowanie terminologii zgodnej z podaną na stronie projektu [19],
- wymagane jest stosowanie się do standardu kodowania podanego na stronie projektu Zend Framework [21].

3.2.3 Standardy kodowania

Standardy kodowania służą ujednoczeniu wyglądu tworzonego kodu. Jest to zbiór zasad, których stosowanie pomaga pilnować odpowiednia konfiguracja środowisk programistycznych, ale których ostateczna weryfikacja i kontrola należy do programistów. Standardy kodowania obejmują wiele aspektów kodu programu, takich jak:

Formatowanie kodu - definiuje szerokość wcięcia, maksymalną długość wiersza oraz liczbę pustych wierszy oddzielających kolejne definicje i deklaracje klas.

Konwencje nazewnicze - określa schemat nadawania nazw funkcjom, klasom, zmiennym itp.

Komentowanie kodu - ujednolica sposób komentowania kodu, rodzaj komentarzy, ich ilość oraz złożoność. Istotne jest także, czy styl komentowania jest zgodny z narzędziem do automatycznego generowania dokumentacji.

Konstrukcje programistyczne - określa, jakie konstrukcje są zalecane, a jakie zabronione dla konkretnych problemów.

Kod powinien być logiczny, spójny i czysty. Powinien nie tylko zapewniać właściwe działanie aplikacji, ale również sam siebie opisywać. Dlatego też należy dbać o odpowiednie nadawanie nazw klasom, funkcjom i zmiennym, aby właściwie i bez dodatkowych komentarzy oddawały intencje programistów.

Stosowanie właściwego standardu kodowania niesie za sobą szereg zalet. Przede wszystkim przyspiesza lokalizowanie błędów w przypadku wykrycia awarii lub niewłaściwego działania programu. Nie bez znaczenia jest też łatwość zrozumienia wyjętego z kontekstu kawałka kodu. Przejrzysty kod przekłada się bezpośrednio na mniejszą ilość błędów. Kluczowa jest także łatwość wdrożenia nowych programistów do pracy przy projekcie. Należy jednak pamiętać, że zbytne przykładanie uwagi do standardów kodowania lub błędne ich zdefiniowanie pociąga za sobą wydłużenie czasu pracy. Często wymagają stosowania dodatkowych struktur, co może redukować kreatywność. Bardzo istotnym problemem jest też świadome ignorowanie standardów przez niektórych programistów.

Standardy kodowania są ważne w każdym projekcie programistycznym, jednak dla projektu, nad którym pracuje większa liczba osób są one niezwykle istotne. Pomagają one w zarządzaniu projektem oraz zmniejszają liczbę błędów, co bezpośrednio przekłada się na lepszą jakość kodu. W realizowanym projekcie standardy kodowania są zdefiniowane dwójako. Wynika to z istnienia dwóch różnych użytkowników końcowych.

- Aplikacja kliencka dla kart konferencyjnych jest oprogramowaniem tworzonym według standardów określonych przez Politechnikę Poznańską. Z uwagi na fakt, że projekt ten jest kontynuacją już istniejącego i zakładany jest jego rozwój, programistom postawiono wymóg stosowania się do specyfikacji zdefiniowanej na stronie projektu [13].
- Standardy kodowania dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii są zgodne ze standardami kodowania dla Zend Framework [21].

3.2.4 Wymagania dotyczące dokumentacji

Dokumentację projektu opracowano zgodnie z zalecaniami Działu Rozwoju Oprogramowania Politechniki Poznańskiej, natomiast jej kształt i forma dodatkowo została uzgodniona z Działem Obsługi i Eksploatacji Politechniki Poznańskiej. W ramach dokumentacji powstały następujące dokumenty:

- dokument przekazania systemu do eksploatacji,
- wyniki testów,
- instrukcja obsługi użytkownika końcowego i administratora,

- instrukcja instalacji,
- instrukcja obsługi RabbitMQ dla administratora,
- dokumentacja usługi sieciowej,
- dokumentacja bazy danych.

Dodatkowym zaleceniem było obszerne stosowanie komentarzy w kodzie źródłowym na etapie tworzenia. Jednak ta forma dokumentacji nie została ujęta w żadne normy, czy standardy, a stanowiła jedynie udogodnienie podczas programowania oraz pomoc w dalszym rozwoju aplikacji.

W przypadku aplikacji dla Wielkopolskiego Centrum Onkologii zalecenia odnośnie dokumentacji zostały dostarczone przez klienta. W ramach umowy z klientem dostarczono dwa dokumenty, tj. instrukcję instalacji oraz instrukcję użytkownika końcowego. W tej części projektu ważną kwestią było także odpowiednie komentowanie kodu. Aplikacja kliencka Wielkopolskiego Centrum Onkologii została bardzo ściśle powiązana z już istniejącym systemem. Należało zatem uwzględnić możliwość dokonywania zmian w kodzie aplikacji lub jej fragmentach przez pracowników Działu Informatyki Wielkopolskiego Centrum Onkologii. Z tego względu dobrze przygotowane komentarze kodu będą bardzo pomocne podczas tego typu prac programistycznych. Forma komentarzy została dopasowana do specyfikacji dostępnej na stronie domowej Zend Framework. Dokumenty wewnętrzne, tj. dostępne dla pracowników Politechniki Poznańskiej zostały przygotowane zgodnie z zalecaniami Działu Rozwoju Oprogramowania Politechniki Poznańskiej, a ich wykaz jest jednakowy z podanym wyżej dla aplikacji klienckiej kart konferencyjnych.

Architektura systemu

System zarządzania kartami elektronicznymi składa się z trzech głównych komponentów:

- Aplikacji centralnej SCMS, która odbiera wszystkie zamówienia na karty z aplikacji klienckich i udostępnia funkcje, umożliwiające administratorowi zarządzanie całym procesem personalizacji i drukowania kart. Dodatkowo od tej wersji systemu aplikacja centralna umożliwia również zarządzanie użytkownikami modułów klienckich kart pracowniczych i konferencyjnych.
- Aplikacji klienckich, umożliwiających wysyłanie zamówień na karty do aplikacji centralnej, śledzenie przebiegu procesu personalizacji kart, a następnie zarządzanie gotowymi kartami.
- Aplikacji komunikacyjnej, która pozwala na przesyłanie komunikatów między modułami klienckimi, a aplikacją centralną [1].

Dodatkowo moduł kliencki dla kart konferencyjnych wymagał uwzględnienia zewnętrznych źródeł danych, z których moduły mogą korzystać w celu ułatwienia procesu składania zamówień na karty. Obecna wersja systemu korzysta z niżej przedstawionych źródeł danych:

- System eLogin - zintegrowany system Politechniki Poznańskiej umożliwiający na uwierzytelnianie pracowników i studentów uczelni. W przypadku modułów klienckich pozwala na logowanie się użytkowników aplikacji kart pracowniczych i konferencyjnych. Jest to dodatkowa funkcja niezależna od użytkowników zdefiniowanych w aplikacji centralnej.
- System eKadry – system wykorzystywany jedynie w aplikacji klienckiej kart pracowniczych w celu pobrania informacji o pracownikach uczelni do lokalnej bazy danych aplikacji. Moduł korzysta z tych danych w celu złożenia zleceń na utworzenie kart przypisanych do wybranych pracowników [1].
- Pliki w formatach CSV, XLS i XLSX - źródło danych wykorzystywane przez nową aplikację kliencką kart konferencyjnych w celu ułatwienia wprowadzania danych podczas tworzenia zamówień na nowe karty.
- Archiwa w formacie ZIP - moduł kliencki kart konferencyjnych umożliwia wykorzystanie archiwów w tym formacie do wczytania obrazów, które mają zostać nadrukowane na zamawianych kartach.

W przypadku aplikacji klienckiej dla Wielkopolskiego Centrum Onkologii źródłem danych jest wewnętrzna baza danych głównej aplikacji Wielkopolskiego Centrum Onkologii. W bazie tej przechowywane są informacje o pracownikach wewnętrznych oraz zewnętrznych organizacji, a także informacje o istniejących w systemie kartach inteligentnych oraz zamówieniach na karty. Dodatkowo w bazie danych przechowywane są informacje o ścieżkach do plików ze zdjęciami pracowników.

W aplikacji przygotowanej dla Wielkopolskiego Centrum Onkologii należy wyróżnić także dodatkowy element systemu jakim jest aplikacja wykonana w technologii Web Service mająca na celu udostępnienie interfejsu służącego do zarządzania certyfikatami użytkowników. Ta część systemu jest bezpośrednio związana z istniejącymi w systemie informacyjnym Wielkopolskiego Centrum Onkologii urzędem certyfikacji i kontrolerem domeny.

Dodatkowymi elementami systemu są zapory przeciwogniowe, które mają za zadanie ograniczyć dostęp do systemu jedynie z zaufanych sieci i dodatkowo blokować potencjalne ataki na serwery sieciowe. Całość architektury została przedstawiona na Rysunku 4.1.

4.1 SCMS

System centralny SCMS składa się z następujących modułów:

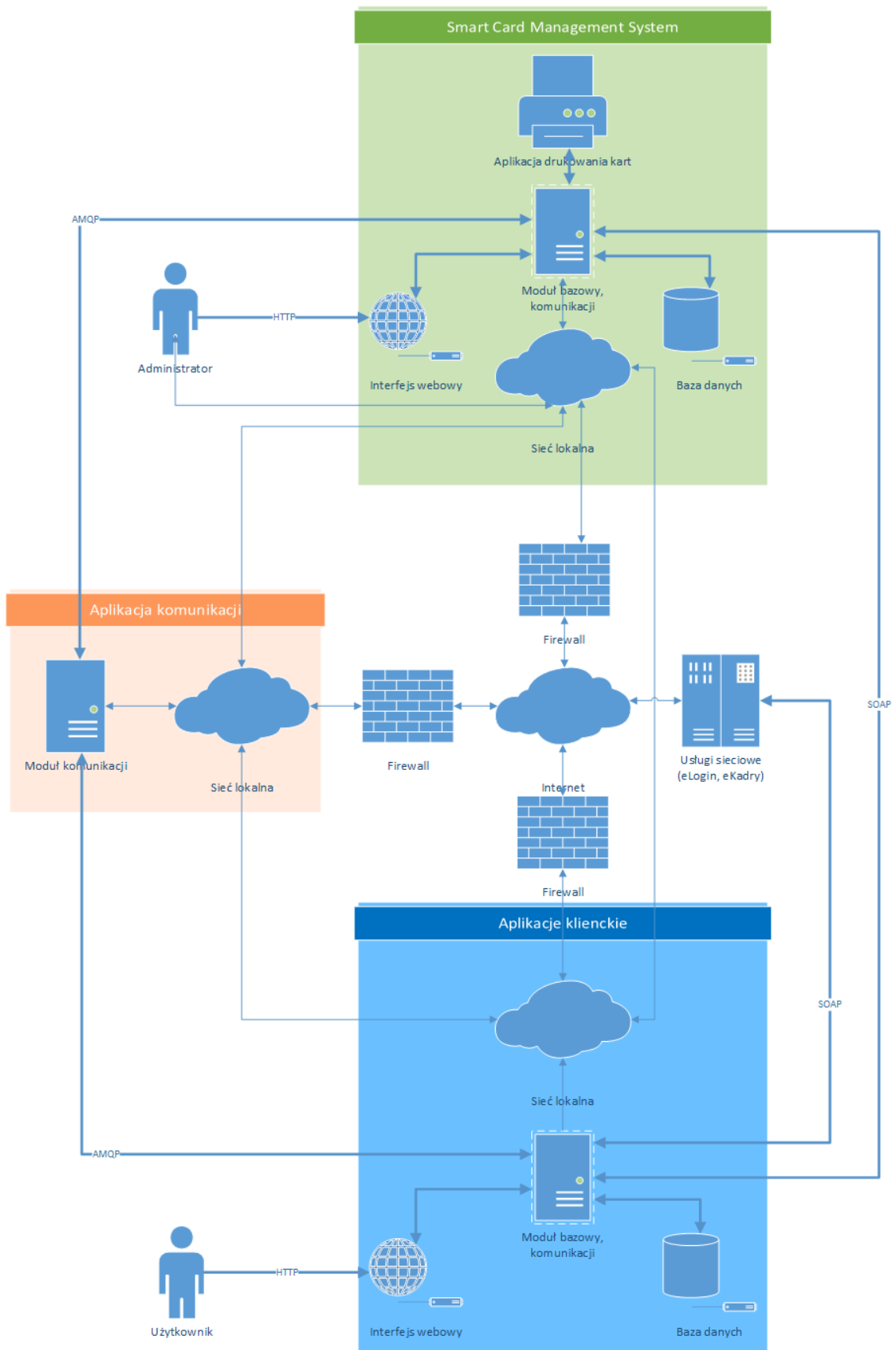
- Moduł bazowy – obsługuje komunikację z bazą danych i logowanie do systemu.
- Moduł administracyjny (interfejs webowy) – jego główne funkcje pozwalają na definiowanie profili kart, blankietów i wzorów graficznych, przydzielanie profili do aplikacji klienckich, zarządzanie zamówieniami na karty przez cały proces personalizacji i definiowanie użytkowników modułów klienckich kart pracowniczych i konferencyjnych.
- Moduł komunikacji – pośredniczy w wymianie komunikatów między modułem centralnym, a modułami klienckimi systemu SCMS.
- Usługi sieciowe – pozwala na udostępnianie metod i obiektów aplikacjom klienckim poprzez wykorzystanie protokołu SOAP. Jego głównym zadaniem jest umożliwienie synchronizacji profili kart w aplikacjach klienckich przy jednoczesnym uwierzytelnianiu ich użytkowników.

Centralny moduł SCMS wykorzystuje bazę danych współdzieloną ze stanowiskami drukującymi. Dzięki temu możliwe jest pobranie informacji o drukowanych partiach kart bezpośrednio z bazy danych przez stacje drukujące i zmianę statusu danej partii po zakończeniu drukowania bez konieczności implementacji dodatkowych metod komunikacji między tymi dwoma elementami systemu [1].

4.2 Aplikacja kliencka konferencji

Z uwagi na wykorzystanie tego samego frameworku i technologii, co w przypadku modułu centralnego SCMS, aplikacja kliencka kart konferencyjnych częściowo składa się z tych samych modułów:

- Moduł bazowy – obsługuje komunikację z bazą danych i logowanie do systemu.



Rysunek 4.1: Architektura systemu

- Moduł administracyjny (interfejs webowy) – umożliwia użytkownikowi na synchronizację przypisanych do niego profili kart i instalację ich w aplikacji klienckiej. Dodatkowo pozwala na ustawienie podstawowych parametrów aplikacji, jak klucza szyfrującego i nazw kolejek RabbitMQ.
- Moduł zarządzania zamówieniami (interfejs webowy) – jest wykorzystywany przez użytkownika do tworzenia nowych zamówień na karty, śledzenia postępu personalizacji już zamówionych kart i zarządzania wydrukowanymi kartami.
- Moduł komunikacji – pośredniczy w wymianie komunikatów między modulem centralnym, a klientem.

Ponadto aplikacja kliencka kart konferencyjnych wykorzystuje osobną bazę danych, której nie współdzielili z pozostałymi elementami systemu.

4.3 Aplikacja kliencka Wielkopolskiego Centrum Onkologii

Aplikacja kliencka dla Wielkopolskiego Centrum Onkologii została wykonana z wykorzystaniem frameworku Zend Framework w wersji 1.11. Z tego względu zmianie uległy jedynie pewne konwencje nazewnictwa stosowane w plikach modułu komunikacyjnego oraz sposób pracy z danymi. Zmiany częściowo były podyktowane także koniecznością wykorzystania starszej wersji PHP w celu zachowania kompatybilności z główną aplikacją Wielkopolskiego Centrum Onkologii. Samą aplikację możemy podzielić na następujące moduły:

- Moduł bazowy - obejmuje komunikację z bazą danych, logowanie do systemu oraz zarządzanie sesją użytkownika.
- Moduł komunikacyjny - zapewnia prawidłową komunikację pomiędzy modulem centralnym, a klientem.
- Moduł kart inteligentnych - zapewnia integrację modułu komunikacyjnego z modulem bazowym. Obejmuje ewidencję kart inteligentnych, wszystkie mechanizmy związane z procesem zamawiania kart inteligentnych oraz umożliwia śledzenie statusu obecnie zatwierdzonych zamówień.
- Moduł administracyjny - pozwala na zmianę ustawień aplikacji. Umożliwia użytkownikowi końcowemu wykonanie synchronizacji profili kart inteligentnych, zmianę ustawień polityki haseł i danych autoryzacyjnych aplikacji oraz edycję danych aktualnie zalogowanego użytkownika.
- Moduł certyfikatów - pozwala na zdalne zarządzanie certyfikatami użytkowników.

Ponadto aplikacja kliencka Wielkopolskiego Centrum Onkologii wykorzystuje osobną, wewnętrzną bazę danych, której nie współdzielili z pozostałymi elementami systemu.

4.4 Komunikacja pomiędzy SCMS a aplikacjami klienckimi

Wymiana komunikatów między elementami systemu jest możliwa na dwa sposoby:

- Wykorzystując aplikację RabbitMQ – aplikacja jest używana głównie do przesyłania informacji o nowych zamówieniach składanych przez użytkowników aplikacji klienckich, a następnie informowania o przebiegu procesu personalizacji. Dodatkowo aplikacje klienckie wykorzystują ten sposób komunikacji do zarządzania odebranymi kartami. Ponadto od tej wersji systemu moduł centralny może korzystać z aplikacji RabbitMQ do zarządzania użytkownikami aplikacji klienckich kart pracowniczych i konferencyjnych. RabbitMQ wykorzystuje protokół AMQP do przesyłania wiadomości do aplikacji komunikacyjnej, która z kolei przypisuje odebrane komunikaty do kolejki aplikacji, do której wiadomość powinna dotrzeć. Wszystkie kolejki są typu FIFO (ang. First In, First Out). Moduły komunikacji poszczególnych elementów systemu są odpowiedzialne za odbieranie wiadomości przechowywanych w kolejkach. Wykorzystanie tego sposobu komunikacji gwarantuje dostarczenie i nie zagubienie wiadomości nawet w przypadku chwilowej awarii aplikacji komunikacyjnej [1].
- Wykorzystując usługę sieciową SOAP – aplikacje klienckie korzystają z tej metody komunikacji głównie do synchronizacji profili kart i uwierzytelniania użytkowników systemu. Dodatkowo klienci kart pracowniczych i konferencyjnych wykorzystują tę metodę przy logowaniu się użytkowników z systemu eLogin, zaś klientowi kart pracowniczych umożliwia ponadto pobieranie danych o pracownikach Politechniki Poznańskiej z systemu eKadry. Ten sposób wymiany danych nie jest niezawodny, lecz był wymagany, aby umożliwić współpracę z uczelnianymi systemami eLogin i eKadry [1].

4.5 Schemat bazy danych

Z uwagi na bardzo zbliżone zastosowania schemat nowego modułu kart konferencyjnych jest podobny do schematu bazy danych modułu kart pracowniczych. Główną różnicą jest brak tabel odpowiedzialnych za przechowywanie informacji o pracownikach Politechniki Poznańskiej, jak również o właścicielu karty. Karty zamawiane przy wykorzystaniu nowego modułu cechują się większą swobodą definiowania możliwych pól znajdujących się na karcie, jak również przeznaczeniem tworzonej karty, więc domyślnie nie posiadają one właścicieli. Dodatkowo schemat bazy danych został rozbudowany o tabelę umożliwiającą ograniczenie dostępności profili, do których mają dostęp dani użytkownicy, a tabela przechowująca informacje o użytkownikach została uzupełniona o kolumnę informującą, czy dany użytkownik został zablokowany przez system.

Prace nad aplikacją kliencką dla Wielkopolskiego Centrum Onkologii wymagały integracji już istniejącego w systemie schematu bazy danych ze schematem aplikacji klienckiej kart pracowniczych. Z tego względu zdecydowano się na utrzymywanie w bazie danych dwóch odrębnych schematów. Jeden z nich został przeznaczony na dane bezpośrednio związane z przetwarzaniem zamówień na karty inteligentne, podczas gdy drugi stanowił

oryginalny schemat głównej aplikacji Wielkopolskiego Centrum Onkologii rozbudowany o nowe obiekty.

Z racji na niewielkie różnice w wymaganiach schemat aplikacji klienckiej kart pracowniczych został tylko częściowo zmieniony. Usunięto tabele skojarzone z danymi pracowników Politechniki Poznańskiej, słownik terminów oraz tabele przechowujące informacje o obecnych w systemie kartach inteligentnych. Dokonano także zmian umożliwiających grupowanie kart w zlecenia oraz dodano atrybut odpowiedzialny za przechowywanie kodu transportowego.

Po stronie oryginalnego schematu głównej aplikacji Wielkopolskiego Centrum Onkologii dodano tabele umożliwiające przechowywanie danych związanych z:

- szablonami filtrów wyszukiwania,
- listami zatwierdzonych zleceń,
- historią zamówień,
- historią zmian statusów kart inteligentnych,
- plikami powiązаныmi z kartami inteligentnymi,
- plikami powiązаныmi z pracownikami Wielkopolskiego Centrum Onkologii,
- certyfikatami użytkowników,
- statusami kart i zleceń,
- danymi użytkowników związanymi z usługą Active Directory.

Ponadto w celu usprawnienia działania części skryptów PHP przygotowano w bazie danych specjalne widoki oraz funkcje pozwalające na pobranie wyłącznie koniecznych z punktu widzenia danego skryptu danych. W celu ujednoczenia widoku listy pracowników utworzono widok pobierający dane zarówno z tabeli pracowników wewnętrznych Wielkopolskiego Centrum Onkologii, jak i pracowników zewnętrznych. Dodatkowo widok ten wykorzystuje funkcję, która dla zadanych argumentów zwraca wszystkie działy, w których jest zatrudniony dany pracownik. Analogiczny widok bazy danych przygotowany został dla widoku listy kart inteligentnych.

Schematy obu nowych modułów zostały załączone w postaci plików na płycie CD.

Opis implementacji

5.1 Narzędzia

Do zarządzania pracą zastosowano system zarządzania projektami Redmine wraz z systemem kontroli wersji Subversion 1.6.17 [7]. Baza danych została zaprojektowana za pomocą narzędzia do zarządzania i rozwoju bazy danych EMS SQL Manager for PostgreSQL. Również dalsze jej modyfikacje były prowadzone za pomocą tej aplikacji. Programowanie odbywało się przy użyciu następujących środowisk programistycznych:

- PhpStorm,
- Eclipse,
- Visual Studio 2012.

Testy funkcjonalne stworzono za pomocą narzędzia do automatyzacji testów Selenium oraz środowiska Eclipse.

Dodatkowo dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii wykorzystano system kontroli wersji Git oraz dodatkowe prywatne repozytorium kodu przechowywane w serwisie Bitbucket. Repozytorium to było dostępne dla grupy programistów odpowiedzialnej za prace nad aplikacją kliencką Wielkopolskiego Centrum Onkologii oraz pracowników Działu Informatyki Wielkopolskiego Centrum Onkologii. Dzięki wykorzystaniu dodatkowego repozytorium zarówno programiści, jak i potencjalni przyszli użytkownicy mieli możliwość sprawdzania zmian dokonywanych w aplikacji klienckiej oraz całym systemie.

Poza wymienionymi wcześniej narzędziami wykorzystano także bezpłatne narzędzie do obsługi wirtualnych maszyn Oracle VM VirtualBox. Zostało ono wykorzystane podczas prac z systemem Windows Server 2012 oraz podczas testów przeprowadzonych na systemach operacyjnych CentOS 6.5 i CentOS 6.4.

5.2 Technologia

Aplikacje zostały zaimplementowane w języku PHP. Aplikacja dla kart konferencyjnych wykorzystuje platformę programistyczną PHPLiteMVC. Na serwerze zainstalowany został interpreter nginx w wersji 1.4.1 wraz z silnikiem proxy oraz modulem https. Obie apli-

kacje korzystają z bazy danych PostgreSQL, która jest wspierana przez platformę programistyczną. W celu zapewnienia niezawodnej komunikacji między serwerem, a modułami klienckimi wykorzystane zostało oprogramowanie RabbitMQ w wersji 3.0.1, który jest zgodny ze standardem protokołu AMQP. Ponadto, aby umożliwić przejrzystą implementację synchronizacji profili kart konferencyjnych, zastosowane zostały usługi internetowe opisane w języku WSDL (ang. Web Services Description Language), wykorzystujące protokół SOAP (ang. Simple Object Access Protocol). Dodatkowo po stronie użytkownika aplikacja kart konferencyjnych wykorzystuje popularną bibliotekę języka JavaScript [3] – jQuery [2] w wersji 1.9. Na niej oparta jest również biblioteka DataTables umożliwiająca przejrzystą i prostą w implementacji wizualizację informacji przechowywanych w bazie danych.

Aplikacja przeznaczona dla Wielkopolskiego Centrum Onkologii różni się wykorzystaną platformą jaką jest Zend Framework w wersji 1.11. Dodatkowo z uwagi na kompatybilność projektowanego rozwiązania z istniejącą częścią systemu wykorzystana została biblioteka jQuery w wersji 1.5 oraz silnik bazy danych PostgreSQL w wersji 8.3. Na serwerze testowym aplikacji klienckiej Wielkopolskiego Centrum Onkologii wykorzystano serwer Apache. W projekcie zaimportowano także bibliotekę PHPExcel [15] wykorzystaną do generowania dokumentów w formacie XLS oraz plug-in DataTables [14] przeznaczony do prezentowania danych w formie tabeli.

Część aplikacji oparta na technologii Web Service oraz platformie .NET [8] została zaimplementowana w języku C# z wykorzystaniem zintegrowanego środowiska programistycznego Microsoft Visual Studio 2012.

5.3 Harmonogram projektu

Prace nad aplikacją kliencką dla Wielkopolskiego Centrum Onkologii zostały podzielone na następujące etapy:

Kamień milowy I - Moduł komunikacyjny i ewidencja kart inteligentnych: 01.12.2013

Kamień milowy II - Zarządzanie certyfikatami użytkowników: 22.12.2013

Realizacja aplikacji kart konferencyjnych nastąpiła w jednym kroku i nie została podzielona na etapy. Zakończenie prac było zaplanowane na 01.12.2013.

5.4 Realizacja zadań

Prace nad systemem przebiegały pod kierownictwem mgr inż. Marka Gosławskiego - kierownika Działu Rozwoju Oprogramowania oraz prof. dr hab. inż. Joanny Józefowskiej. Dodatkowo z projektem związany był zespół menedżerów:

inż. Mateusz Jarmużek - Informatyka - Technologie Wytwarzania Oprogramowania,
Wydział Informatyki, Politechnika Poznańska

inż. Tomasz Wagner - Informatyka - Technologie Wytwarzania Oprogramowania, Wydział Informatyki, Politechnika Poznańska (kierownik projektu)

inż. Mateusz Wolańczyk - Informatyka - Technologie Wytwarzania Oprogramowania, Wydział Informatyki, Politechnika Poznańska

Kilka pierwszych spotkań z zespołami menedżerów i programistów miało na celu określenie zakresu prac, zapoznanie zespołów z już istniejącym systemem oraz określenie funkcji, o które system miał zostać wzbogacony. Późniejsze cotygodniowe spotkania dotyczyły postępów prac, napotkanych problemów oraz służyły dokonywaniu drobnych zmian w podziale zadań między programistów. Dodatkowo odbyło się kilka spotkań z przedstawicielami Wielkopolskiego Centrum Onkologii, których celem było zapoznanie programistów z już istniejącym tam systemem, doprecyzowanie wymagań użytkowników względem systemu oraz prezentacja kolejnych gotowych części aplikacji klienckiej. Podczas ostatniego spotkania z pracownikami Działu Informatyki Wielkopolskim Centrum Onkologii wyznaczonego na dzień 28 stycznia 2014 roku przeprowadzono testy manualne całej aplikacji oraz dokonano ostatecznych zmian w projekcie.

Prace programistyczne poprzedzone zostały zapoznaniem się z wykorzystywanymi technologiami oraz platformami programistycznymi. Późniejsze problemy i pytania natury technicznej spotykały się z rzetelnym wsparciem pracowników Działu Rozwoju Oprogramowania. Zadania menedżerów ograniczały się do przygotowania opowieści użytkownika (ang. user stories) oraz przypadków użycia, jak również przydziału programistów do zadań. Komunikacja pomiędzy członkami zespołu projektowego wspierana była przez środowisko Redmine, a współbieżne prace programistyczne możliwe były dzięki korzystaniu z systemu kontroli wersji Subversion. Wszystkie te technologie znacznie ułatwiły prace nad projektem.

5.5 Testy

Bardzo istotnym elementem procesu wytwarzania oprogramowania jest przeprowadzenie testów. Dzięki nim możliwe jest zapewnienie odpowiedniej jakości produktu końcowego. Sam proces testowania składa się głównie z testów jednostkowych i funkcjonalnych. Jednak w przypadku aplikacji internetowych dodatkowymi składnikami odpowiedzialnymi za jakość są czas odpowiedzi oraz zużycie zasobów pamięciowych. Dlatego też przeprowadza się testy wydajnościowe mające na celu określenie wartości tych parametrów. Wzajemne inspekcje kodu przez programistów oraz błędy i uwagi zgłaszane na bieżąco przez kadrę nadzorującą i klientów również są istotnym elementem testów, które dodatkowo pozwalają na szybką reakcję i przyspieszają realizację projektu.

W ramach realizowanego projektu przeprowadzone zostały testy funkcjonalne. Są one tworzone w celu ustalenia, czy program spełnia założenia i oczekiwania klienta. Realizuje się je bez wiedzy na temat budowy programu, a wyłącznie na bazie wymagań funkcjonalnych.

Najlepsze efekty podczas tworzenia tego rodzaju testów osiąga się, gdy osoba projektującą instancje testowe nie jest zaangażowana w programowanie. Dlatego też testy zostały zaprojektowane przez kadrę menedżerską. Ich zrealizowaniem zajęli się programiści, którzy za pomocą środowiska Selenium dokonali automatyzacji według specyfikacji instancji

testowych.

Środowisko Selenium jest oprogramowaniem zintegrowanym z przeglądarką internetową Firefox. Pozwala ono na tworzenie automatycznych testów, ich ręczną edycję oraz zapisywanie. Dodatkowo możliwe jest wyeksportowanie testu Selenium do języka Java w formie gotowego kodu junit. W takiej postaci możliwa jest jego dalsza edycja oraz uruchomienie w środowisku programistycznym obsługującym język Java. Na potrzeby projektu wykorzystane zostało środowisko Eclipse.

Testy umożliwiły na wykrycie błędów, które nie zostały zauważone podczas pisania klienta kart konferencyjnych i modyfikowania centralnej aplikacji SCMS. Dzięki temu udało się zwiększyć niezawodność modułu klienckiego i zwiększyć ergonomię korzystania z już istniejących elementów po stronie serwera.

W przypadku aplikacji dla Wielkopolskiego Centrum Onkologii przewidziano wyłącznie testy manualne. Odbywały się one równolegle do procesu wytwarzania oprogramowania. Każdy kolejny element systemu był testowany przez zespół programistów oraz jednocześnie przez przedstawiciela Działu Informatyki Wielkopolskiego Centrum Onkologii. Takie podejście pozwoliło przyspieszyć przepływ informacji od klienta do zespołu programistów, ponieważ każda zmiana w systemie była oceniana na bieżąco. Dodatkowo programiści mogli prezentować propozycje swoich rozwiązań poprzez przedstawienie działającego fragmentu aplikacji użytkownikowi końcowemu, co znacznie usprawniło proces dopasowywania rozwiązania do wymagań stawianych przez klienta.

Ostateczne testy aplikacji zostały przeprowadzone dnia 28 stycznia 2014 roku w siedzibie Wielkopolskiego Centrum Onkologii.

5.6 Dokumentacja

Dokumentacja bazy danych

Zawiera opis wszystkich tabel i znajdujących się w nich kolumn wraz z definicją relacji między poszczególnymi tabelami. Dokumentacja została wykonana dla obu modułów klienckich, jak również dla serwera SCMS z uwagi na zmiany, jakie zostały wprowadzone w bazie danych serwera podczas realizacji projektu.

Dokumentacja usługi sieciowej

Opis metod i struktur obiektów technologii Web Service wykorzystywanej przez system SCMS do uwierzytelniania klientów i synchronizacji profili kart.

Dokumentacja instalacji

Zawiera dokładną instrukcję instalacji modułu kart konferencyjnych w systemie operacyjnym CentOS w wersji 6.4, jak również instrukcję instalacji aplikacji klienckiej dla Wielkopolskiego Centrum Onkologii w systemie operacyjnym CentOS w wersji 6.5.

Dokumenty zawierają:

- komendy niezbędne do instalacji wymaganych przez aplikacje pakietów,
- konfigurację bazy danych,
- opis ustawień RabbitMQ,
- ustawienia Web Service'ów,
- instrukcje pierwszego uruchomienia aplikacji.

Dodatkowo instrukcja instalacji klienta Wielkopolskiego Centrum Onkologii jest uzupełniona o opis niezbędnych zmian w obecnym kodzie aplikacji SWPDI (System Wspomagania Pracy Działu Informatyki) i przygotowanie niezbędnej struktury katalogów dla nowego modułu. W instrukcji zawarto także dokładne instrukcje konfiguracji usługi certyfikatów wykonanej w technologii Web Service, która współpracuje bezpośrednio z urzędem certyfikacji oraz kontrolerem domeny. Dla tych elementów także zawarto informacje o rekomendowanych ustawieniach.

Instrukcja instalacji RabbitMQ

Instrukcja opisująca proces instalacji serwera RabbitMQ po stronie aplikacji centralnej w celu poprawnego odbierania wiadomości od modułów klienckich.

Instrukcje obsługi użytkownika końcowego i administratora

Zawiera opis funkcji systemu SCMS znajdującego się po stronie serwera, jak i modułu klienckiego kart konferencyjnych. Instrukcje zostały wykonane w dwóch wersjach: dla użytkownika i dla administratora systemu.

Dla aplikacji Wielkopolskiego Centrum Onkologii przygotowano jeden dokument pełniący rolę instrukcji obsługi użytkownika końcowego. Dokument ten zawiera informacje dotyczące modułu kart inteligentnych oraz dodatkowych funkcji wymaganych przez klienta.

Dokumenty przekazania modułów systemu do eksploatacji

Zawierają krótki opis systemu SCMS, klienta Wielkopolskiego Centrum Onkologii i kart konferencyjnych wraz z określeniem krytyczności poszczególnych modułów i ich przeznaczenia. Każdy dokument zawiera ponadto opis logicznej zależności między modułami systemu, spis wykorzystanych technologii i minimalne wymagania sprzętowe. Dodatkowo dokumenty zawierają wyniki testów funkcjonalnych dla każdego z modułów.

Wyniki testów

Dokument zawiera podsumowanie wszystkich testów wykonanych dla nowych modułów klienckich.

Powyższe dokumenty można znaleźć na dołączonej do pracy płycie CD.

5.7 Podsumowanie realizacji

Prace implementacyjne nad projektem rozpoczęły się w październiku 2013 roku i trwały cztery miesiące. Realizacja aplikacji kart konferencyjnych przebiegała planowo, a niewielkie opóźnienia poszczególnych zadań względem harmonogramu nie wpłynęły na końcowy rezultat prac.

Realizacja zadań związanych z aplikacją dla Wielkopolskiego Centrum Onkologii rozpoczęła się z opóźnieniem względem harmonogramu. Było to konsekwencją konieczności ustalenia wspólnego terminu spotkania mającego na celu zaprezentowanie obecnego systemu znajdującego się po stronie Wielkopolskiego Centrum Onkologii oraz przekazanie odpowiednich plików niezbędnych do podjęcia prac programistycznych. Największe opóźnienie wynikało z konieczności dostosowania istniejącego już modułu komunikacyjnego do wymagań stawianych przez Zend Framework oraz architekturę głównej aplikacji klienta. To zagadnienie okazało się zdecydowanie niedoszacowane. Ostatecznie wszystkie zadania udało się zakończyć w planowanym terminie.

Finalnie wszystkie zagadnienia ujęte w harmonogramie prac zostały w pełni zrealizowane. System został wdrożony w Wielkopolskim Centrum Onkologii, a moduł kart konferencyjnych przekazany do użytku i dalszego rozwoju Działowi Rozwoju Oprogramowania Politechniki Poznańskiej.

Wnioski i zalecenia wdrożeniowe

6.1 Plan wdrożenia oprogramowania

Wdrożenie aplikacji nie było częścią tej pracy dyplomowej. Po zakończeniu procesu implementacji, kody źródłowe wraz z całą wiedzą merytoryczną zostaną przekazane do Działu Rozwoju Oprogramowania.

W przypadku aplikacji klienckiej dla Wielkopolskiego Centrum Onkologii przewiduje się obecność programisty odpowiedzialnego za jej przygotowanie podczas wdrożenia nowego systemu. Jest to podyktowane dużą złożonością zaimplementowanego rozwiązania i koniecznością konfiguracji systemu. Wdrożenie nie jest częścią pracy dyplomowej, dlatego jego przygotowanie oraz przebieg będzie uwarunkowany zaleceniami Działu Rozwoju Oprogramowania oraz Działu Informatyki Wielkopolskiego Centrum Onkologii.

6.2 Uwagi dotyczące użytkowania systemu

W celu poprawnego działania zaprojektowanych rozwiązań konieczne jest zapewnienie wsparcia dla technologii JavaScript przez wykorzystywane przeglądarki internetowe. Dodatkowo w celu zapewnienia poprawnej szaty graficznej aplikacji oraz prawidłowego działania wykorzystanych komponentów zaleca się wykonanie aktualizacji obecnych w systemie przeglądarek do ich najnowszych wersji.

Aplikacja kliencka Wielkopolskiego Centrum Onkologii została przygotowana z uwzględnieniem wymagań stawianych przez klienta dotyczących wersji wykorzystywanych bibliotek oraz narzędzi. Wykorzystywane biblioteki oraz narzędzia objęte ograniczeniami wersji zaprezentowane są poniżej:

- Zend Framework 1.11.11,
- PHP 5.2.6,
- PostgreSQL 8.3,
- jQuery 1.6.

W celu zapewnienia poprawnego działania aplikacji zaleca się stosowanie wyżej wymienionych bibliotek oraz narzędzi w wersjach równych lub wyższych od zaprezentowanych.

Bibliografia

- [1] P. Brzeziński, K. Czyżnielewski, D. Minc, M. Sobkowiak, *System zarządzania cyklem życia kart elektronicznych*, Politechnika Poznańska, Poznań, 2013
- [2] E. Castledine, C. Sharkie, *jQuery - od nowicjusza do wojownika ninja*, Helion, Gliwice, 2012
- [3] D. Crockford, *JavaScript. Mocne strony*, Helion, Gliwice, 2009
- [4] Dokumentacja protokołu komunikacyjnego, Politechnika Poznańska, [dostęp: 5 stycznia 2014], <<https://conaiten.put.poznan.pl/attachments/2719/amqp.pdf>>
- [5] *Karta kryptograficzna Comarch SmartCard*, Comarch, [dostęp: 5 stycznia 2014], <<https://www.comarch.pl/finanse/produkty/comarch-smartcard>>
- [6] Kevin J. O'Brien, *Smart card: Invented here*, The New York Times, 10 sierpnia 2005 [dostęp: 5 stycznia 2014], <<http://www.nytimes.com/2005/08/09/world/europe/09iht-card.html>>
- [7] D. Merkel, *PHP5. Narzędzia dla ekspertów*, Helion, Gliwice, 2011
- [8] T. Northrup, M. Snell, *Web Applications Development with Microsoft .NET Framework 4*, Microsoft Press, Redmond, 2010
- [9] Oferta firmy Control System FMN, [dostęp: 5 stycznia 2014], <[http://www.cs.pl/92,Karty plastikowe](http://www.cs.pl/92,Karty%20plastikowe)>
- [10] Opis narzędzia certutil, [dostęp: 5 stycznia 2014], <<http://technet.microsoft.com/en-us/library/cc732443.aspx>>
- [11] *Smart Card Security*, Smart Card Basics, [dostęp: 5 stycznia 2014], <<http://www.smartcardbasics.com/smart-card-security.html>>
- [12] Standardy kodowania dla aplikacji kart konferencyjnych, Politechnika Poznańska, [dostęp: 5 stycznia 2014], <<https://conaiten.put.poznan.pl/projects/templates/wiki/>>
- [13] Standardy kodowania w języku PHP, Politechnika Poznańska, [dostęp: 5 stycznia 2014], <[http://conaiten.put.poznan.pl/projects/templates/wiki/Standard_kodowania_w_języku_PHP](http://conaiten.put.poznan.pl/projects/templates/wiki/Standard_kodowania_w_j%C4%99zyku_PHP)>

- [14] Strona główna projektu DataTables, [dostęp: 5 stycznia 2014], <<http://www.datatables.net>>
- [15] Strona główna projektu PHPExcel, [dostęp: 5 stycznia 2014], <<http://phpexcel.codeplex.com>>
- [16] Strona główna projektu TCPDF, [dostęp: 5 stycznia 2014], <<http://www.tcpdf.org>>
- [17] A. Szelaǳ, *Windows Server 2008. Infrastruktura klucza publicznego (PKI)*, Helion, Gliwice, 2008
- [18] M. Szymański, *Smart Cards (ICC)*, styczeń 2003 [dostęp: 5 stycznia 2014], <http://students.mimuw.edu.pl/SO/Projekt02-03/temat4-g2/mirosław_szymanski/sc.htm>
- [19] Wykaz terminologii wykorzystywanej w aplikacjach, Politechnika Poznańska, [dostęp: 5 stycznia 2014], <<https://conaiten.put.poznan.pl/projects/card-management/wiki/Nazewnictwo>>
- [20] *XML Web Services Overview (Przegląd serwisów sieciowych w XML)*, Microsoft, [dostęp: 5 stycznia 2014], <[http://msdn.microsoft.com/en-us/library/w9fdtx28\(v=aspnet.11\).aspx](http://msdn.microsoft.com/en-us/library/w9fdtx28(v=aspnet.11).aspx)>
- [21] *Zend Framework Coding Standard for PHP (Standardy kodowania Zend Framework dla PHP)*, Zend Framework 2, [dostęp: 5 stycznia 2014], <<http://framework.zend.com/manual/1.12/en/coding-standard.html>>

Dodatki

Przypadki użycia

Wymagania funkcjonalne dla aplikacji centralnej

WF-SCMS2-MCP-ADM-12 - Wyświetlanie realizowanych zamówień

Aktor: ADMIN-1

Priorytet: wysoki

Główny scenariusz:

1. ADMIN-1- otwiera okno prezentujące zamówienia do realizacji.
2. System wyświetla obecne zamówienia.

Rozszerzenia:

- 1.A. Brak istniejących zamówień w systemie.
- 1.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-MCP-ADM-13 - Generacja podglądu wydruku kart

Aktor: ADMIN-1

Priorytet: wysoki

Warunki początkowe:

1. USER-1- za pośrednictwem aplikacji klienckiej złożył zamówienie na karty przesyłając wymagane dane (WF-SCMS2-MCP-USER-10).

Główny scenariusz:

1. Wyświetlenie zamówień (WF-SCMS-MCP-ADM-12).
2. ADMIN-1 wybiera partię kart.
3. ADMIN-1 wybiera opcję wygenerowania podglądu.
4. System generuje podgląd wydruku kart.

WF-SCMS2-MCP-ADM-14 - Zlecenie kart do wydruku

Aktor: ADMIN-1

Priorytet: wysoki

Główny scenariusz:

1. ADMIN-1 wyświetla zamówienia (WF-SCMS2-MCP-ADM-12).
2. ADMIN-1 wybiera opcję wydruku kart.
3. System drukuje karty.

Rozszerzenia:

- 1.B. ADMIN-1- edytuje zamówienie (np. usuwa konkretne karty).
- 1.B.1. Przejście do kroku 2.

WF-SCMS2-MCP-ADM-15 - Zgłoszenie gotowych kart

Aktor: ADMIN-1

Priorytet: wysoki

Warunki początkowe:

1. ADMIN-1 wydrukował karty (WF-SCMS2-MCP-ADM-14) z konkretnego zamówienia.

Główny scenariusz:

1. ADMIN-1 wyświetla zamówienia (WF-SCMS2-MCP-ADM-12).
2. ADMIN-1 wybiera opcję potwierdzenia zrealizowania zamówienia.
3. System wysyła informację do aplikacji klienckiej o zrealizowaniu zamówienia.

Rozszerzenia:

- 3.A. System wysyła informację na adres e-mail klienta.

WF-SCMS2-MCP-ADM-7 - Wyświetlenie historii zamówień

Aktor: ADMIN-1

Priorytet: wysoki

Główny scenariusz:

1. ADMIN-1 otwiera okno prezentujące historię zamówień.
2. System wyświetla historię zamówień.

Rozszerzenia:

- 2.A. Brak istniejących historycznych zamówień w systemie.
 - 2.A.1. System wyświetla odpowiedni komunikat.
- 2.B. ADMIN-1 wybiera historię konferencji.
 - 2.B.1. System prezentuje listę kart, partii oraz zamówień danej konferencji.
- 2.C. ADMIN-1 wybiera historię zamówienia.
 - 2.C.1. System prezentuje listę kart oraz partii, które znajdowały się w zamówieniu.
- 2.D. ADMIN-1 wybiera historię partii.
 - 2.D.1. System prezentuje listę kart, które znajdowały się w partii.
- 2.E. ADMIN-1 wybiera historię karty.
 - 2.E.1. System prezentuje kartę.
- 2.F. ADMIN-1 ma możliwość przechodzenia przez całą hierarchię w dowolny sposób, np. z konkretnej karty do zamówienia, które zawierało tę kartę i odwrotnie.

WF-SCMS2-MCP-ADM-8 - Przeszukiwanie historii zamówień

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

1. ADMIN-1 wyświetla historię zamówień (WF-SCMS2-MCP-ADM-7).
2. ADMIN-1 wpisuje słowa kluczowe w wyszukiwarce.
3. System filtruje historię zamówień.
4. System wyświetla wynik filtrowania.

Rozszerzenia:

- 4.A. Pusty wynik filtrowania, System wyświetla stosowny komunikat, powrót do kroku 2.

WF-SCMS2-MCP-ADM-16 - Konstrukcja profilu dla aplikacji klienckiej

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

1. ADMIN-1 wyświetla dostępne profile.
2. ADMIN-1 wybiera opcję dodania profilu.
3. ADMIN-1 konfiguruje profil.
4. ADMIN-1 przydziela profil do konkretnej aplikacji klienckiej.

Rozszerzenia:

- 3.A. Profil już istnieje.
- 3.A.1. ADMIN-1 przydziela istniejący profil do konkretnej aplikacji klienckiej.

WF-SCMS2-MCP-ADM-17 - Blokowanie i odblokowanie użytkowników

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

1. ADMIN-1 wyświetla dostępnych użytkowników.
2. ADMIN-1 wybiera opcję zablokowania użytkownika.
3. ADMIN-1 potwierdza operację zablokowania użytkownika.
4. System zapisuje informację.

Rozszerzenia:

- 2.A. ADMIN-1 wybiera opcję odblokowania zablokowanego użytkownika.
- 2.A.1. ADMIN-1 potwierdza operację odblokowania użytkownika.
- 2.A.2. System zapisuje informację.

Wymagania funkcjonalne dla aplikacji klienckiej kart konferencyjnych

WF-SCMS2-MCP-1 - Logowanie do systemu

Aktor: ANONYMOUS-1

Priorytet: wysoki

Główny scenariusz:

1. ANONYMOUS-1 podaje swój login i hasło.
2. ANONYMOUS-1 loguje się do systemu.
3. System uwierzytelnia podane dane.
4. System loguje użytkownika jako USER-1.
5. System udostępnia profile przeznaczone dla USER-1.

Rozszerzenia:

- 3.A. Login lub hasło są niepoprawne.
 - 3.A.1. System wyświetla odpowiedni komunikat.
 - 3.A.2. Powrót do kroku 1.
- 1.A. ANONYMOUS-1 ma możliwość zalogowania przez eLogin.

WF-SCMS2-MCP-USER-9 - Wprowadzanie danych

Aktor: USER-1

Priorytet: wysoki

Główny scenariusz:

1. USER-1 otwiera okno tabeli danych.
2. System wyświetla ekran tabeli wczytanych danych.
3. USER-1 wybiera opcję wczytania danych z pliku.
4. System weryfikuje poprawność wczytanych danych.
5. System wyświetla stosowny komunikat.

Rozszerzenia:

- 3.A. USER-1 ma możliwość wprowadzenia danych ręcznie do wiersza tabeli.
- 4.A. Weryfikacja zakończona niepowodzeniem.
 - 4.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-MCP-USER-10 - Zamawianie kart

Aktor: USER-1

Priorytet: wysoki

Warunki początkowe:

1. Wprowadzone wcześniej dane (WF-SCMS2-MCP-USER-9).

Główny scenariusz:

1. USER-1 wybiera profil z listy przypisanych do niego profili.
2. USER-1 otwiera okno tabeli danych.
3. System wyświetla tabelę danych.
4. USER-1 wybiera pojedynczy rekord z danymi.
5. USER-1 wybiera opcję złożenia zamówienia.
6. Podgląd kart.
7. USER-1 potwierdza złożenie zamówienia.

Rozszerzenia:

- 3.A. Brak wprowadzonych danych.
- 3.A.1. System wyświetla odpowiedni komunikat.
- 4.A. USER-1 ma możliwość zaznaczenia wielu rekordów z danymi.
- 4.B. USER-1 ma możliwość wybrania profilu dla konkretnej karty.
- 6.A. USER-1 ma możliwość wywołania Edycji karty (WF-SCMS2-MCP-USER-21).

WF-SCMS2-MCP-USER-22 - Wydruk PDF z zamówieniem

Aktor: USER-1

Priorytet: średni

Warunki początkowe:

- 1. Zamawianie kart (WF-SCMS2-MCP-USER-10).

Główny scenariusz:

- 1. USER-1 otwiera podgląd zamówień (WF-SCMS2-MCP-USER-10).
- 2. USER-1 wybiera opcję wydruku zamówienia.
- 3. System generuje plik PDF z danymi zamówienia.
- 4. System rozpoczyna proces drukowania pliku.

WF-SCMS2-MCP-USER-11 - Podgląd realizowanych zamówień

Aktor: USER-1

Priorytet: wysoki

Warunki początkowe:

- 1. Złożone wcześniej zamówienie (WF-SCMS2-MCP-USER-10).

Główny scenariusz:

- 1. USER-1 otwiera okno realizowanych zamówień.

Rozszerzenia:

- 1.A. Brak zamówień.
- 1.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-MCP-USER-21 - Edycja kart

Aktor: USER-1

Priorytet: wysoki

Główny scenariusz:

- 1. System wyświetla pola tekstowe do modyfikacji danych.
- 2. USER-1 ma możliwość edytowania określonych danych na karcie.

Rozszerzenia:

- 2.A. Krok może zostać powtórzony dla dowolnej liczby kart dowolną ilość razy.

WF-SCMS2-MCP-USER-18 - Podgląd profili

Aktor: USER-1

Priorytet: wysoki

Główny scenariusz:

- 1. USER-1 otwiera podgląd profili.
- 2. System wyświetla podgląd profili.

WF-SCMS-MCP-USER-23 - Podgląd historycznych zamówień

Aktor: USER-1

Priorytet: średni

Główny scenariusz:

1. USER-1 otwiera podgląd historycznych zamówień.
2. System prezentuje podgląd historycznych zamówień.

Rozszerzenia:

- 2.A. Brak istniejących historycznych zamówień w systemie.
 - 2.A.1. System wyświetla odpowiedni komunikat.
- 2.B. USER-1 wybiera historię zamówienia.
 - 2.B.1. System prezentuje listę kart.
- 2.C. USER-1 wybiera historię karty.
 - 2.C.1. System prezentuje kartę.
- 2.E. USER-1 ma możliwość przechodzenia przez całą hierarchę w dowolny sposób, np. z konkretnej karty do zamówienia, które zawierało tą kartę i odwrotnie.

Wymagania funkcjonalne dla aplikacji klienckiej Wielkopolskiego Centrum Onkologii

WF-SCMS2-WCO-ADM-2 - Zlecenie wydania kart

Aktor: ADMIN-1

Priorytet: wysoki

Główny scenariusz:

1. ADMIN-1 otwiera zakładkę zlecenia wydania kart.
2. ADMIN-1 wybiera pracownika lub gościa, typ karty wybierany jest automatycznie.
3. System pobiera zdjęcie pracownika lub gościa z bazy danych i dodaje do zlecenia.
4. System prezentuje podgląd karty.
5. ADMIN-1 potwierdza zlecenie.
6. System weryfikuje poprawność wczytanych danych.
7. System wysyła zlecenie do SCMS.
8. System zapisuje informacje.

Rozszerzenia:

- 3.A. Brak zdjęcia wybranego pracownika lub gościa w bazie danych.
 - 3.A.1. System wyświetla odpowiedni komunikat.
- 4.A. Edycja danych karty przez ADMIN-1 po zapoznaniu się z podglądem.
 - 4.A.1. System prezentuje zaktualizowany podgląd karty.
 - 4.A.1.A ADMIN-1 wybiera opcję zastosowania zmian do wszystkich kart.
 - 4.A.1.A.1. System prezentuje zaktualizowane podglądy wszystkich kart.
- 6.A. Weryfikacja zakończona niepowodzeniem.
 - 6.A.1. System wyświetla odpowiedni komunikat.
- 7.A. Błąd umieszczenia zlecenia w kolejce.
 - 7.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-WCO-ADM-4 - Wgrywanie certyfikatu dla karty

Aktor: ADMIN-1

Priorytet: wysoki

Warunki początkowe:

1. Złożone wcześniej zlecenie (WF-SCMS2-WCO-ADM-2).

Główny scenariusz:

1. ADMIN-1 otwiera zakładkę listy kart.
2. ADMIN-1 wybiera kartę bez wgranego certyfikatu.
3. ADMIN-1 potwierdza aktywację karty.
4. System wysyła żądanie wygenerowania certyfikatu do CA.

Rozszerzenia:

- 2.A. Brak kart bez wgranego certyfikatu.
 - 2.A.1. System wyświetli odpowiedni komunikat.
- 4.A. Generowanie certyfikatów zakończone błędem.
 - 4.A.1. System wyświetla odpowiedni komunikat i pyta się, czy żądanie ma zostać ponownie wysłane.
 - 4.A.1.A. ADMIN-1 decyduje się na ponowienie żądania.
 - 4.A.1.A.1 Powrót do kroku 4.

WF-SCMS2-WCO-ADM-6 - Generowanie raportu

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

1. ADMIN-1 otwiera zakładkę generowania raportu.
2. ADMIN-1 wybiera szablon raportu.
3. ADMIN-1 wybiera przedział czasowy, dla którego ma być sporządzony raport.
4. ADMIN-1 wybiera dane, które mają się znaleźć w raporcie.
5. ADMIN-1 potwierdza generowanie raportu.
6. System generuje raport.
7. System wyświetla raport.

Rozszerzenia:

- 7.A. ADMIN-1 wybiera opcję zapisu raportu do pliku.
 - 7.A.1. ADMIN-1 wybiera nazwę, lokację i format pliku.
 - 7.A.2. System zapisuje raport do pliku.

WF-SCMS2-WCO-ADM-7 - Zmiana ewidencji kart

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

1. ADMIN-1 otwiera zakładkę listy kart.
2. ADMIN-1 wybiera kartę.
3. ADMIN-1 zmienia status karty.
4. ADMIN-1 zatwierdza zapis zmian.
5. System zapisuje informacje.

Rozszerzenia:

- 3.A. ADMIN-1 dołącza plik do karty.
- 3.A.1. ADMIN-1 wybiera nazwę i lokalizację pliku.
- 3.A.2. System kopiuje wskazany plik.

WF-SCMS2-WCO-ADM-8 - Podgląd realizowanych zleceń

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

- 1. ADMIN-1 otwiera zakładkę listy zleceń.
- 2. ADMIN-1 wybiera zlecenie.
- 3. System wyświetla dane zlecenia.

WF-SCMS2-WCO-ADM-9 - Powiadomienia na pulpicie użytkownika systemu SWPDI (dashboard)

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

- 1. System otrzymuje informację o zdarzeniu (zmiana statusu zlecenia, wygasający certyfikat dla karty).
- 2. System wyświetla powiadomienie na pulpicie SWPDI.
- 3. ADMIN-1 wybiera powiadomienie.
- 4. System wyświetla szczegółowe informacje o zdarzeniu.

WF-SCMS-WCO-ADM-10 - Anulowanie zlecenia

Aktor: ADMIN-1

Priorytet: średni

Warunki początkowe:

- 1. Złożone wcześniej zlecenia (WF-SCMS2-WCO-ADM-2) dostępne w podglądzie zleceń (WF-SCMS2-WCO-ADM-8).

Główny scenariusz:

- 1. ADMIN-1 wybiera zlecenie.
- 2. ADMIN-1 wybiera opcję anulowania zlecenia.
- 3. ADMIN-1 potwierdza anulowanie zlecenia.
- 4. System usuwa zlecenie z kolejki SCMS.

Rozszerzenia:

- 4.A. Informacja o braku możliwości usunięcia zlecenia z kolejki.
- 4.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-WCO-ADM-12 - Potwierdzenie odbioru zlecenia

Aktor: ADMIN-1

Priorytet: wysoki

Warunki początkowe:

- 1. Złożone wcześniej zlecenia (WF-SCMS2-WCO-ADM-2) dostępne w podglądzie zleceń (WF-SCMS2-WCO-ADM-8).

Główny scenariusz:

1. ADMIN-1 wybiera zlecenie.
2. ADMIN-1 wybiera opcję potwierdzenia odbioru.
3. ADMIN-1 potwierdza zapis danych.
4. System zapisuje informacje.
5. System przesyła potwierdzenie odbioru do SCMS.

Rozszerzenia:

- 6.A. Błąd wysyłania potwierdzenia do SCMS.
- 6.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-WCO-ADM-13 - Widok listy pracowników

Aktor: ADMIN-1

Priorytet: wysoki

Główny scenariusz:

1. ADMIN-1 otwiera widok listy pracowników.
2. System prezentuje listę pracowników.
3. ADMIN-1 ma możliwość filtrowania i sortowania pracowników.

Rozszerzenia:

- 2.A. Brak pracowników w bazie danych.
- 2.A.1. System wyświetla odpowiedni komunikat.

WF-SCMS2-WCO-ADM-14 - Widok zrealizowanych zleceń

Aktor: ADMIN-1

Priorytet: wysoki

Główny scenariusz:

1. ADMIN-1 otwiera widok zrealizowanych zleceń.
2. System prezentuje spójny widok zrealizowanych i aktywnych zleceń z możliwością filtrowania.

Rozszerzenia:

- 2.A. Brak zrealizowanych zleceń.
- 2.A.1. System wyświetla odpowiedni komunikat.
- 2.B. ADMIN-1 ma możliwość podejrzenia listy kart składających się na zlecenie.

WF-SCMS2-WCO-ADM-15 - Słownik dla statusów

Aktor: ADMIN-1

Priorytet: średni

Główny scenariusz:

1. ADMIN-1 otwiera zakładkę "Słownik statusów".
2. System prezentuje dodatkowy słownik dla statusów.
3. ADMIN-1 może edytować słownik.

Płyta CD

- Dokument przekazania do eksploatacji aplikacji kart konferencyjnych
przekazanie_do_eksploatacji_karty_konferencyjne.pdf
- Dokument przekazania do eksploatacji aplikacji Wielkopolskiego Centrum Onkologii
przekazanie_do_eksploatacji_wco.pdf
- Dokumentacja bazy danych aplikacji kart konferencyjnych
dokumentacja_bazy_danych_karty_konferencyjne.pdf
- Dokumentacja bazy danych aplikacji Wielkopolskiego Centrum Onkologii
dokumentacja_bazy_danych_wco.pdf
- Dokumentacja protokołu komunikacyjnego AMQP
dokumentacja_amqp.pdf
- Dokumentacja usługi sieciowej
dokumentacja_usluga_sieciowa.pdf
- Elektroniczna wersja pracy dyplomowej w formacie PDF
moduly_dodatkowe_dla_systemu_zarzadzania_karta_elektroniczna_scms.pdf
- Elektroniczna wersja pracy dyplomowej w formacie TEX
moduly_dodatkowe_dla_systemu_zarzadzania_karta_elektroniczna_scms.tex
- Informacja o dostępności kodu źródłowego aplikacji klienckiej Wielkopolskiego Centrum Onkologii
aplikacja_wco.txt
- Instrukcja instalacji aplikacji kart konferencyjnych
instrukcja_instalacji_karty_konferencyjne.pdf
- Instrukcja instalacji aplikacji Wielkopolskiego Centrum Onkologii
instrukcja_instalacji_wco.pdf
- Instrukcja obsługi aplikacji kart konferencyjnych
instrukcja_obslugi_karty_konferencyjne.pdf
- Instrukcja obsługi aplikacji Wielkopolskiego Centrum Onkologii
instrukcja_obslugi_wco.pdf
- Instrukcja obsługi RabbitMQ
instrukcja_obslugi_rabbitmq.pdf

- Kod źródłowy aplikacji klienckiej kart konferencyjnych
kod_zrodlowy_karty_konferencyjne.zip
- Komplet materiałów składających się na treść pracy dyplomowej
moduly_dodatkowe_dla_systemu_zarzadzania_karta_elektroniczna_scms.zip
- Kopia karty pracy dyplomowej
karta_pracy.pdf
- Plik konfiguracyjny RabbitMQ dla aplikacji kart konferencyjnych
rabbit_config_karty_konferencyjne.json
- Schemat bazy danych aplikacji kart konferencyjnych
schemat_bazy_danych_karty_konferencyjne.tar