

Inżynieria oprogramowania

**Projekt systemu informatycznego
System obsługi salonu kosmetycznego**

Raport wykonalności

Oliwia Rusinek

Rok akademicki 2025/2026

Wersja 2.0, 16.05.2026

*Zakres aktualizacji: moduł klientek, moduł pracowników, moduł wizyt, rekordy wizyt, statusy
aktywna/odbyta/anulowana.*

1. Założenia wstępne

Zleceniodawcą opracowania jest niewielki salon kosmetyczny świadczący usługi pielęgnacyjne i upiększające. Dokument przygotowano jako raport wykonalności projektu systemu informatycznego dla przedmiotu Inżynieria Oprogramowania. Podstawą opracowania są wymagania prowadzącego, wykonana aplikacja demonstracyjna oraz diagramy UML dopasowane do aktualnego zakresu systemu.

Celem projektu jest wdrożenie aplikacji webowej wspomagającej pracę recepcjonistki. System ma uporządkować obsługę klientek, pracowników wykonujących zabiegi oraz rezerwacji wizyt. W aktualnej wersji aplikacja posiada lewy panel modułów, osobny moduł klientek, osobny moduł pracowników oraz moduł wizyt działający jak tabela rekordów.

1.1. Historia wersji

Wersja	Data	Opis zmian
1.0	15.05.2026	Pierwsza wersja dokumentacji i aplikacji z podstawową rejestracją wizyt.
2.0	16.05.2026	Aktualizacja dokumentacji do nowej aplikacji: moduły po lewej stronie, pracownicy, zabiegi, rekordy wizyt, statusy aktywna/odbyta/anulowana.

1.2. Zakres projektu

- logowanie recepcjonistki do systemu,
- dodawanie, edycja i usuwanie klientek,
- dodawanie, edycja i usuwanie pracowników salonu,
- rejestrowanie wizyt z wyborem klientki, pracownika, zabiegu, daty, godziny i ceny,
- wyświetlanie wizyt jako czytelnych rekordów w tabeli,
- automatyczne rozróżnianie statusu wizyty: aktywna, odbyta lub anulowana,
- zapisywanie danych w pamięci przeglądarki localStorage.

2. Charakterystyka firmy

Salon kosmetyczny jest małą firmą usługową. Główna działalność polega na wykonywaniu zabiegów kosmetycznych takich jak manicure hybrydowy, pedicure, stylizacja rzęs, henna brwi, makijaż okolicznościowy oraz zabiegi pielęgnacyjne twarzy. Klientki umawiają się na konkretne godziny i do wybranych lub dostępnych pracowników.

Recepcjonistka odpowiada za kontakt z klientkami, rejestrację wizyt oraz prowadzenie terminarza. Pracownicy salonu wykonują zabiegi i posiadają swoje specjalizacje. Właściciel nadzoruje pracę salonu, organizację grafików i jakość obsługi.

Struktura organizacyjna i moduły systemu



Rysunek 1. Struktura organizacyjna i moduły systemu

2.1. Procesy biznesowe

- rejestracja nowej klientki w systemie,
- dodanie pracownika wykonującego zabiegi,
- wybór usługi oraz terminu wizyty,
- przypisanie wizyty do klientki i pracownika,
- monitorowanie statusu wizyty,
- anulowanie lub usunięcie błędnie dodanej wizyty,
- przegląd historii wizyt odbytych.

3. Opis stanu obecnego

Przed wdrożeniem systemu salon prowadził terminarz w sposób tradycyjny, np. w zeszycie, kalendarzu papierowym albo w prostym arkuszu kalkulacyjnym. Taki sposób pracy jest wystarczający tylko przy bardzo małej liczbie wizyt. Wraz ze wzrostem liczby klientek pojawiają się problemy organizacyjne.

3.1. Infrastruktura sprzętowa i środowisko IT

Element	Stan obecny	Uwagi
Stanowisko recepcji	1 komputer lub laptop	Wystarczające do uruchomienia aplikacji w przeglądarce.
Internet	Dostęp podstawowy	W aktualnej wersji aplikacja działa lokalnie. Internet może być potrzebny w przyszłości.
Oprogramowanie	Przeglądarka internetowa	System działa w przeglądarce, bez instalacji.
Dane	Papier / arkusz / pliki lokalne	Brak centralnego porządku i historii zmian.

3.2. Problemy obecnego sposobu pracy

Problem	Skutek dla salonu
Wszystkie informacje w jednym miejscu bez podziału na moduły	Recepcjonistka trudniej odnajduje konkretne dane.
Brak historii wcześniejszych wizyt	Nie widać, które wizyty były wczoraj lub przedwczoraj.
Brak listy pracowników	Nie wiadomo od razu, kto wykonuje dany zabieg.
Brak statusów wizyt	Trudno odróżnić wizytę aktywną od odbytej lub anulowanej.
Ręczne prowadzenie terminarza	Ryzyko pomyłki przy terminach i przypisaniu osoby.

4. Charakterystyka użytkowników

Podstawowym użytkownikiem systemu jest recepcjonistka. Może to być osoba z podstawową znajomością komputera, dlatego aplikacja powinna być maksymalnie czytelna. Nowa wersja aplikacji posiada panel boczny z modułami: Panel główny, Klientki, Pracownicy i Wizyty. Dzięki temu użytkownik nie widzi wszystkich formularzy naraz.

Użytkownik	Zakres pracy w systemie	Poziom IT	Wymagania interfejsu
Recepcjonistka	Dodawanie klientek, pracowników i wizyt	Podstawowy/średni	Czytelne moduły, rekordy w tabelach, proste przyciski.
Właściciel salonu	Podgląd organizacji pracy i liczby wizyt	Podstawowy/średni	Panel podsumowujący, szybki wgląd w aktywne wizyty.
Pracownik salonu	Informacja o przypisanych wizytach	Podstawowy	Czytelne dane: kto, kiedy, na jaki zabieg.

5. Wymagania na nowy system

5.1. Wymagania funkcjonalne

ID	Wymaganie	Priorytet	Realizacja w aplikacji
F1	Logowanie recepcjonistki loginem i hasłem	wysoki	Formularz logowania.
F2	Osobny moduł klientek	wysoki	Zakładka Klientki w lewym panelu.
F3	Dodawanie, edycja i usuwanie klientek	wysoki	Formularz i tabela klientek.
F4	Osobny moduł pracowników	wysoki	Zakładka Pracownicy.
F5	Dodawanie, edycja i usuwanie pracowników	wysoki	Formularz pracownika i tabela pracowników.
F6	Osobny moduł wizyt	wysoki	Zakładka Wizyty.
F7	Dodanie wizyty z wyborem klientki, pracownika, zabiegu, daty i godziny	wysoki	Formularz wizyty.
F8	Wyświetlanie wizyt jako rekordów	wysoki	Tabela wizyt z kolumnami.

F9	Edycja, anulowanie i usuwanie wizyt	wysoki	Przyciski przy każdym rekordzie.
F10	Automatyczny status odbyta dla wizyt z przeszłości	średni	Funkcja obliczania statusu na podstawie daty i godziny.
F11	Zapis danych po odświeżeniu strony	średni	localStorage przeglądarki.

5.2. Wymagania pozafunkcjonalne

- czas reakcji systemu powinien być natychmiastowy przy typowej liczbie danych salonu,
- interfejs powinien być czytelny dla recepcjonistki bez zaawansowanej wiedzy informatycznej,
- system powinien działać w przeglądarce internetowej,
- dane powinny pozostać po odświeżeniu strony,
- system powinien dać się rozbudować o bazę danych i logowanie wieloużytkownikowe.

5.3. Wymagania dziedzinowe

- wizyta musi posiadać datę, godzinę, klientkę, pracownika i zabieg,
- pracownik powinien mieć specjalizację, aby łatwiej dobrać osobę do usługi,
- system powinien umożliwiać historię wizyt odbytych,
- anulowana wizyta powinna pozostać w historii, o ile nie zostanie usunięta,
- termin aktywnej wizyty danego pracownika nie powinien się dublować.

5.4. Wymagania interfejsu użytkownika

Wymaganie	Uzasadnienie
Lewy panel modułów	Użytkownik szybko wybiera obszar pracy.
Osobne tabele rekordów	Wizyty, klientki i pracownicy nie mieszają się na jednym ekranie.
Czytelne statusy kolorami	Recepcjonistka od razu widzi wizyty aktywne, odbyte i anulowane.
Formularze w ramach właściwego modułu	Dodawanie danych jest intuicyjne.
Panel główny ze statystykami	Ułatwia szybkie rozeznanie w bieżącej pracy.

6. Opis proponowanego systemu

Proponowany system jest aplikacją webową uruchamianą w przeglądarce. Aplikacja została wykonana w jednym pliku HTML z kodem CSS i JavaScript. Dane zapisywane są w localStorage, co oznacza, że po odświeżeniu strony nadal widoczne są dodane klientki, pracownicy i wizyty. Jest to rozwiązanie odpowiednie dla wersji demonstracyjnej na laboratorium, a jednocześnie dobrze pokazuje docelową logikę działania systemu.

6.1. Architektura logiczna

Warstwa	Technologia	Rola w systemie
Prezentacji	HTML + CSS	Wyświetlanie panelu bocznego, formularzy i tabel.
Logiki aplikacji	JavaScript	Dodawanie, edycja, usuwanie, obliczanie statusów, walidacja terminów.
Danych	localStorage	Lokalne przechowywanie klientek, pracowników i wizyt w przeglądarce.

6.2. Porównanie stanu obecnego z proponowanym systemem

Obszar	Stan obecny	Proponowany system
--------	-------------	--------------------

Klientki	Zapis ręczny lub arkusz	Osobny moduł klientek z tabelą.
Pracownicy	Brak osobnej ewidencji	Osobny moduł pracowników i specjalizacji.
Wizyty	Terminarz papierowy	Tabela rekordów wizyt.
Status wizyty	Ustalany ręcznie	Automatycznie aktywna/odbyta oraz ręcznie anulowana.
Historia	Trudna do odtworzenia	Widoczne wcześniejsze wizyty.
Czytelność	Wszystko w jednym miejscu	Lewy panel i podział na moduły.

6.3. Bezpieczeństwo systemu

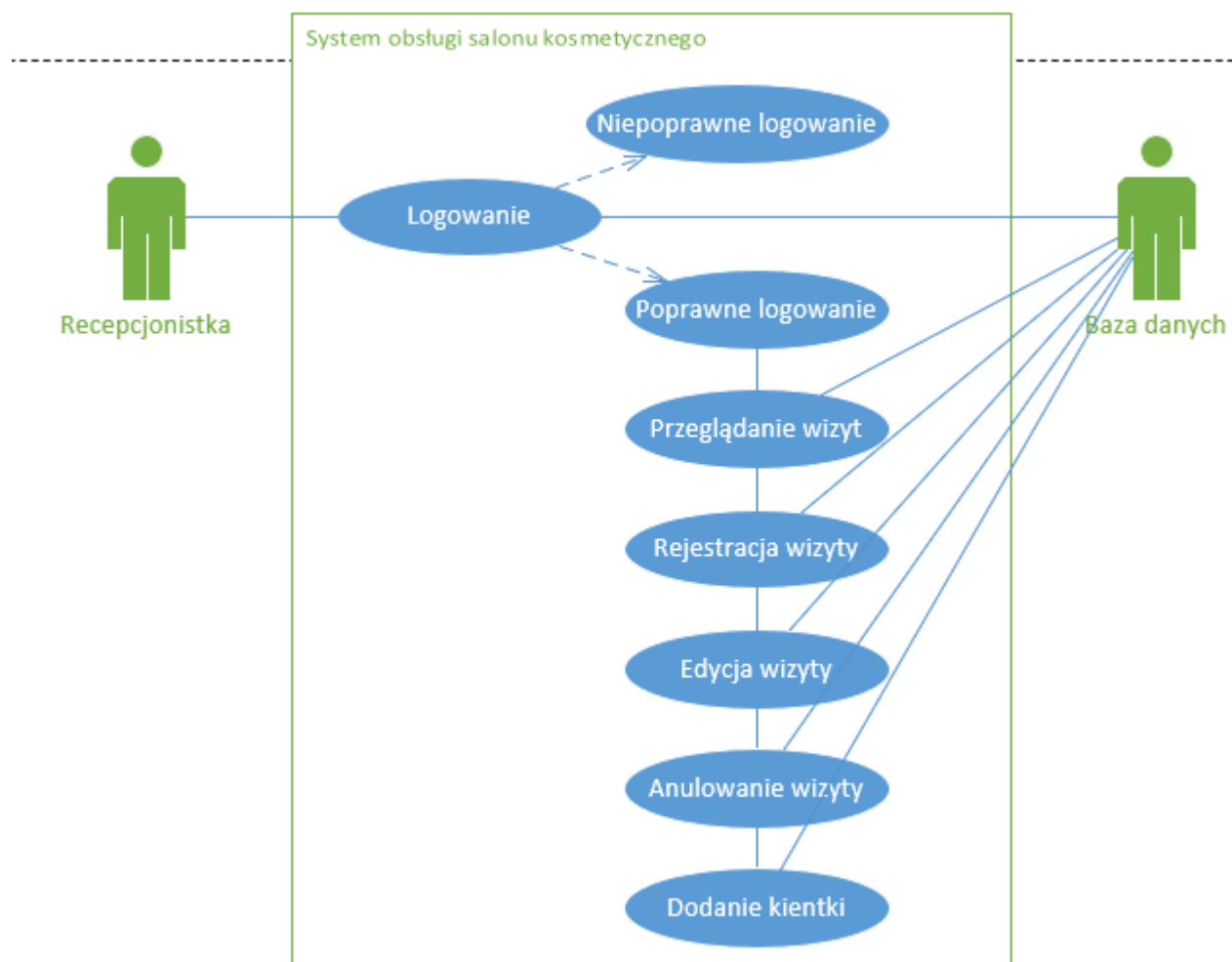
System posiada prosty formularz logowania. W wersji laboratoryjnej login i hasło są zapisane w kodzie, co jest akceptowalne do prezentacji prototypu, ale nie powinno być stosowane w wersji produkcyjnej. W przyszłości należy przenieść dane użytkowników do bazy danych i zastosować szyfrowanie haseł.

- autoryzacja dostępu do panelu aplikacji,
- ograniczenie widoczności danych przed zalogowaniem,
- możliwość rozbudowy o bazę danych i kopie zapasowe,
- możliwość dodania ról użytkowników: recepcja, administrator, właściciel.

7. Diagramy UML i modele systemu

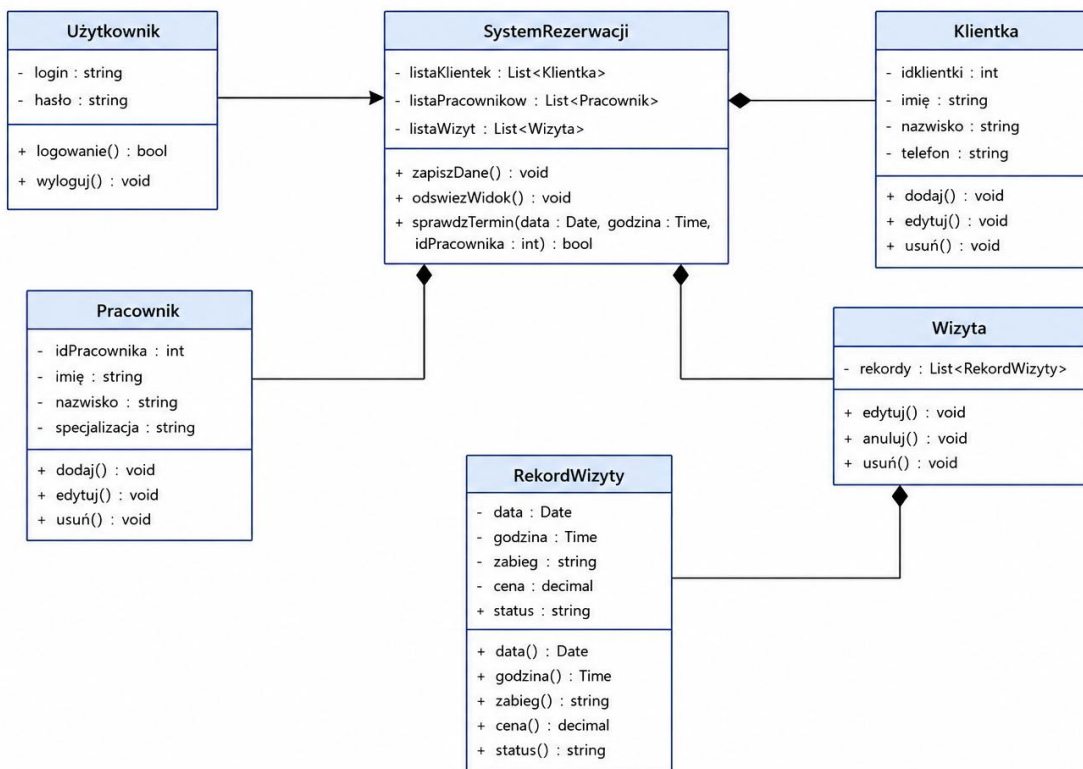
Diagramy w poprzedniej wersji dokumentacji nie były już w pełni zgodne z aktualną aplikacją, ponieważ nie pokazywały modułu pracowników, zabiegów oraz nowego układu wizyt jako rekordów. Poniższe

diagramy zostały zaktualizowane do obecnej wersji aplikacji.

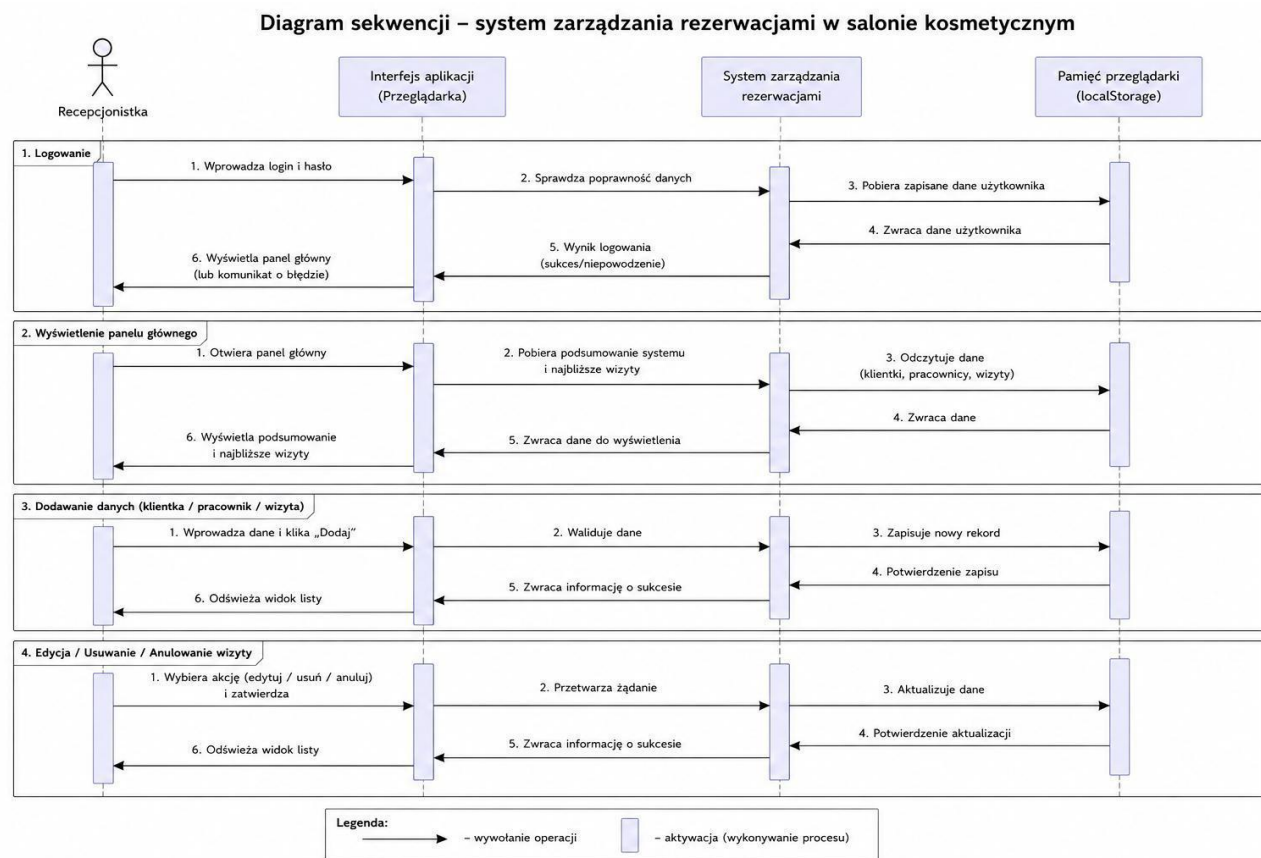


Rysunek 2. Diagram przypadków użycia aktualnej aplikacji

Diagram klas – aktualna wersja aplikacji



Rysunek 3. Diagram klas aktualnej aplikacji

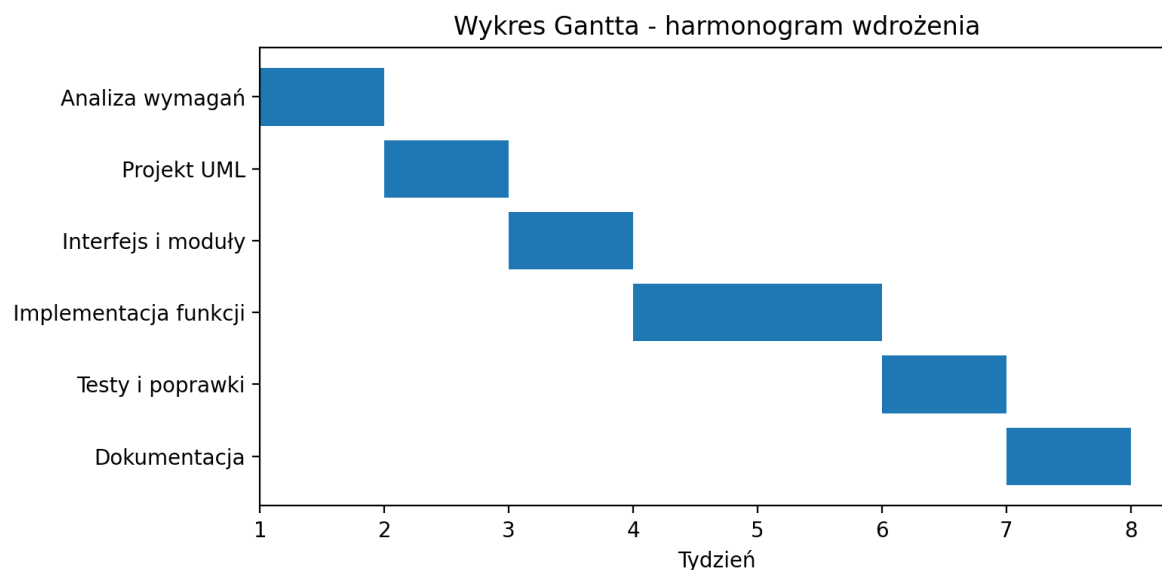


Rysunek 4. Diagram sekwencji dodania wizyty

8. Harmonogram prac

Realizację projektu zaplanowano w cyklu ośmiotygodniowym. Harmonogram obejmuje analizę wymagań, projekt UML, wykonanie modułów, testy oraz przygotowanie dokumentacji. Wdrożenie ma charakter przyrostowy: najpierw powstają podstawowe dane, potem moduł wizyt, a na końcu poprawki czytelności i testy.

Etap	Zakres prac	Czas	Tydzień
1	Analiza wymagań i uwag prowadzącego	1 tydzień	1
2	Projekt UML i struktury danych	1 tydzień	2
3	Projekt interfejsu z panelem modułów	1 tydzień	3
4	Implementacja modułów klientek, pracowników i wizyt	2 tygodnie	4-5
5	Testy dodawania, edycji, usuwania i statusów	1 tydzień	6
6	Dokumentacja i przygotowanie prezentacji	1 tydzień	7
7	Poprawki końcowe	1 tydzień	8



Rysunek 5. Wykres Gantta harmonogramu wdrożenia

9. Koszty i zyski

9.1. Koszty wdrożenia

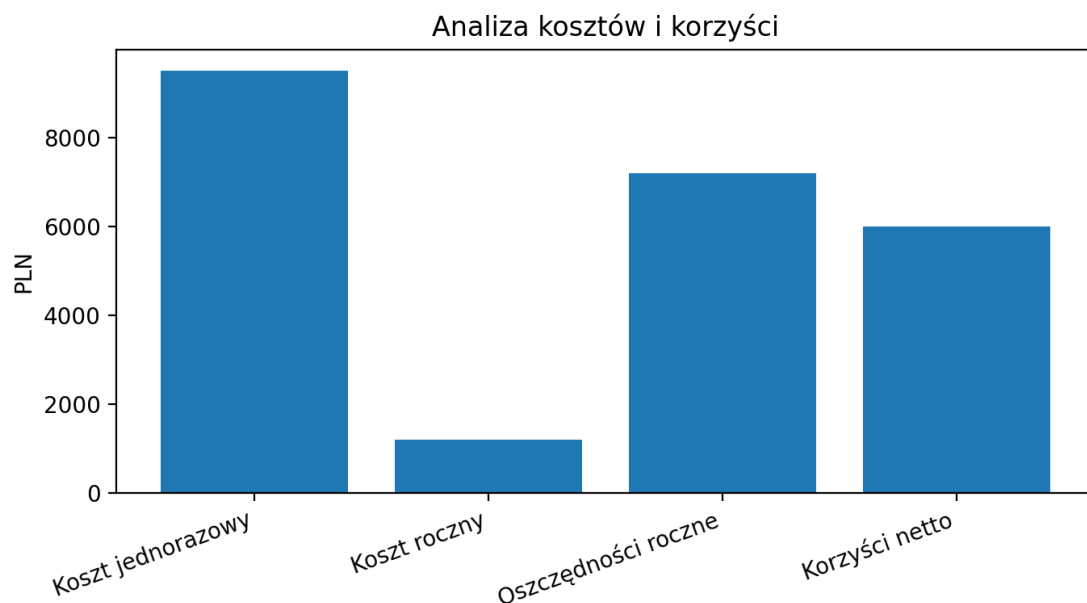
Kategoria kosztu	Opis	Szacowany koszt PLN
Sprzęt	Wykorzystanie istniejącego komputera, ewentualna modernizacja stanowiska	2500
Wytworzenie aplikacji	Przygotowanie prototypu i modułów aplikacji	4500
Testy i poprawki	Sprawdzenie działania formularzy, statusów i zapisu danych	1000
Szkolenie recepcji	Instruktaż obsługi modułów	500
Dokumentacja	Raport wykonalności i opis systemu	1000
RAZEM	Koszty jednorazowe	9500

9.2. Koszty utrzymania

Kategoria	Opis	Koszt roczny PLN
Wsparcie techniczne	Drobne poprawki i aktualizacje	800
Kopie zapasowe / nośnik	Przygotowanie eksportu lub kopii danych	200
Utrzymanie stanowiska	Konserwacja komputera	200
RAZEM	Koszty roczne	1200

9.3. Korzyści ekonomiczne i pozaekonomiczne

Korzyść	Opis	Szacowana wartość roczna PLN
Oszczędność czasu recepcji	Szybsze wyszukiwanie wizyt i klientów	3000
Mniej pomyłek w terminarzu	Kontrola aktywnych terminów pracownika	1500
Lepsza organizacja pracy	Widoczny pracownik, zabieg i status wizyty	1700
Mniej dokumentów papierowych	Ograniczenie zeszytów i kartek	1000
RAZEM	Łączne korzyści roczne	7200



Rysunek 6. Analiza kosztów i korzyści

Przy kosztach jednorazowych około 9500 PLN i rocznych korzyściach netto około 6000 PLN zwrot z inwestycji może nastąpić po około 1,5-2 latach. W przypadku projektu laboratoryjnego najważniejszą korzyścią jest jednak pokazanie kompletnego modelu pracy recepcji oraz czytelnego podziału aplikacji na moduły.

10. Ewolucja systemu

System może być rozwijany stopniowo. Obecna wersja działa jako prototyp, ale struktura danych pozwala na przeniesienie logiki do bazy danych i aplikacji serwerowej.

Kierunek rozwoju	Opis	Horyzont czasowy
Baza danych	Zastąpienie localStorage bazą MySQL lub PostgreSQL	kolejny etap
Konta użytkowników	Role: recepcjonistka, właściciel, administrator	kolejny etap
Filtrowanie wizyt	Filtrowanie po dacie, pracowniku, statusie i zabiegu	krótkoterminowo
Powiadomienia SMS/e-mail	Przypomnienia dla klientek	średnioterminowo
Rezerwacje online	Możliwość samodzielnej rezerwacji przez klientkę	długoterminowo
Raporty	Statystyki najpopularniejszych zabiegów i obłożenia pracowników	długoterminowo

11. Testy systemu

Przypadek testowy	Dane wejściowe	Oczekiwany rezultat	Status
Logowanie poprawne	admin / 1234	Otwarcie panelu głównego	zaliczony
Dodanie klientki	Imię, nazwisko, telefon	Rekord pojawia się w tabeli klientek	zaliczony
Dodanie pracownika	Imię, nazwisko, specjalizacja	Pracownik pojawia się na liście wyboru wizyty	zaliczony
Dodanie wizyty	Klientka, pracownik, zabieg, data, godzina	Rekord pojawia się w tabeli wizyt	zaliczony

Wizyta z przeszłości	Data wcześniejsza niż obecna	Status wyświetla się jako odbyta	zaliczony
Anulowanie wizyty	Kliknięcie Anuluj	Status zmienia się na anulowana	zaliczony
Konflikt terminu	Ten sam pracownik, ta sama data i godzina	System blokuje aktywną wizytę	zaliczony

12. Wnioski

Po aktualizacji aplikacja i dokumentacja są spójne. System nie jest już prostą listą formularzy na jednej stronie, lecz posiada czytelny układ modułowy. Recepcjonistka może osobno zarządzać klientkami, pracownikami i wizytami. Wizyty są przedstawione jako rekordy zawierające datę, godzinę, klientkę, pracownika, zabieg, cenę i status. Taki sposób prezentacji spełnia uwagi dotyczące przejrzystości i pozwala łatwo ocenić, kto, kiedy i na jaki zabieg jest zapisany.

Zaleca się kontynuację projektu i traktowanie obecnej wersji jako poprawnego prototypu systemu obsługi salonu kosmetycznego. Wersja demonstracyjna nadaje się do prezentacji na zajęciach, a po rozbudowie o bazę danych mogłaby zostać wykorzystana w małym salonie usługowym.

13. Spis rysunków

- Rysunek 1. Struktura organizacyjna i moduły systemu
- Rysunek 2. Diagram przypadków użycia aktualnej aplikacji
- Rysunek 3. Diagram klas aktualnej aplikacji
- Rysunek 4. Diagram sekwencji dodania wizyty
- Rysunek 5. Wykres Gantta harmonogramu wdrożenia
- Rysunek 6. Analiza kosztów i korzyści

14. Spis tabel

- Tabela 1. Historia wersji
- Tabela 2. Zakres infrastruktury
- Tabela 3. Problemy obecnego sposobu pracy
- Tabela 4. Charakterystyka użytkowników
- Tabela 5. Wymagania funkcjonalne
- Tabela 6. Wymagania dotyczące interfejsu
- Tabela 7. Architektura logiczna
- Tabela 8. Porównanie stanu obecnego z proponowanym systemem
- Tabela 9. Harmonogram wdrożenia
- Tabela 10. Koszty wdrożenia
- Tabela 11. Koszty utrzymania
- Tabela 12. Korzyści ekonomiczne i pozaekonomiczne
- Tabela 13. Kierunki rozwoju
- Tabela 14. Przypadki testowe