

Plan projektu

(System umawiania wizyt w salonie kosmetycznym)

1.	Cel projektu	3
2.	Organizacja projektu	3
3.	Struktura organizacyjna	3
4.	Role i odpowiedzialność	3
5.	Harmonogram projektu	6
6.	Kosztorys projektu	3
7.	Zasoby projektu	7
8.	Standardy i narzędzia w projekcie	7
9.	Procesy zarządzania	3
a.	zarządzania konfiguracją	3
b.	zarządzania ryzykiem	3
c.	zarządzania testami	3

1. Wstęp

System dotyczy rejestrowania wizyt przez klientów posiadających konta w lokalnym systemie. Użytkownik po zalogowaniu do aplikacji ma możliwość zarezerwowania terminu poprzez przejście w odpowiedni widok. Użytkownik również posiada dostęp do wszystkich przeszłych i przyszłych terminów rezerwowanych wizyt. System powinien umożliwiać użytkownikowi możliwość logowania do aplikacji na urządzeniu dostępnym w lokalu. Użytkownik również powinien mieć dostęp do kosmetyków oferowanych przez salon. Istotne jest również zapewnienie administratorowi łatwego dostępu do informacji o wszystkich klientach.

1. Cel projektu

Celem aplikacji jest umożliwianie klientom salonu kosmetycznego rejestracji na wizyty w salonie. Możliwe jest to po zalogowaniu na konto użytkownika na urządzeniu dostępnym w salonie. Na daną godzinę w danym dniu możliwe jest ustalenie tylko jednej wizyty przez danego użytkownika. Główną funkcjonalnością systemu jest usprawnienie i automatyzacja procesu rejestracji na wizyty w salonie kosmetycznym.

2. Role i odpowiedzialność

Project manager(P_M)

- Przygotowanie harmonogramów działania,
- Nadzorowanie projektu,
- Przydzielanie zadań do realizacji,
- Określanie priorytetów zadań
- Rozwiązywanie problemów,
- Zarządzanie wszystkimi fazami projektu.

Product analyst(P_A):

- Nadzorowanie i monitorowanie wydajności systemu,
- Opracowywanie i prezentowanie strategii dotyczących produktu oraz innowacji,
- Tworzenie narzędzi i skal mierzących poziom satysfakcji klientów,
- Analizowanie danych marketingowych i określanie strategii marketingowych.

Risk manager(R_M):

- Określanie płaszczyzn potencjalnych zagrożeń,
- Opracowywanie planów reagowania na wystąpienie czynników zagrażających,
- Określanie budżetu dla minimalizowania i przeciwdziałania potencjalnym czynnikom ryzykownym,
- Szkolenie pracowników z procesów i standardów mających zapobiegać wystąpieniu czynników ryzyka,
- Określanie prawdopodobieństwa ryzyka i ich wag.

Quality manager(Q_M):

- Określanie oczekiwań i potrzeb klientów,
- Opracowywanie wymagań spełniających potrzeby klientów,
- Zarządzanie jakością produktu,

- Określanie poziomu jakości produktu poprzez wykonywanie odpowiednich testów,

Test manager(T_M):

- Zarządza procesem testowania,
- Zarządza budżetem przeznaczonym na testy,
- Określa strategię testowania,
- Dokumentuje wyniki testów,
- Nadzoruje zespoły testowe.

Programista(P):

- Umiejętność sprawnego posługiwania się określonymi narzędziami programistycznymi,
- Znajomość języków programowania używanych w projekcie,
- Wykonywanie unittestów,
- Pisanie kodu źródłowego aplikacji,
- Znajomość standardów formatowania kodu, używanych w firmie,
- Nanosi poprawki w kodzie aplikacji.

System analyst(S_A):

- Analiza i ocena produkowanego systemu,
- Identyfikacja wymagań systemowych,
- Określanie dodatkowych wymagań jakie powinien spełniać system,
- Identyfikowanie i dokumentowanie zmian w systemie.

Tester(T):

- Wykonywanie testów według procedury testowania,
- Raportowanie wykrytych błędów w bazie danych błędów,

Test designer(T_D):

- Określanie potencjalnych problemów w oprogramowaniu,
- Pisanie testcase-ów,
- Testowanie obciążeniowe aplikacji,
- Korzystanie z narzędzi do automatyzacji testowania aplikacji.

UI designer(UI_D):

- Współpraca z analitykami,
- Tworzenie elementów graficznego interfejsu użytkownika,

Application developer(A_D):

- Określanie metod implementacji wymagań klienta,
- Ocena istniejącego kodu aplikacji,
- Poszukiwanie rozwiązań zwiększających wydajność aplikacji.

Database designer(DB_D):

- Tworzenie dokumentacji bazy danych,
- Tworzenie plików bazy danych,
- Dbanie o bezpieczeństwo bazy danych,

Tabela. Dokumenty wypełniane przez osoby pełniące poszczególne role

Artefakt zadania	Rola w projekcie realizująca zadanie
Plan projektu; Opis wersji	P_M
Harmonogram projektu	P_M
Kosztorys	P_M
Plan testów	T_M
Plan ryzyka	R_M
Plan kontroli jakości	Q_M
Plan zarządzania	P_M
Specyfikacja wymagań	P_A
	P_A
Analiza specyfikacji wymagań	S_A
Diagram bazy danych	DB_D
Diagram interfejsu	UI_D

3. Harmonogram projektu

Planowany początek projektu: 18/03/2022;

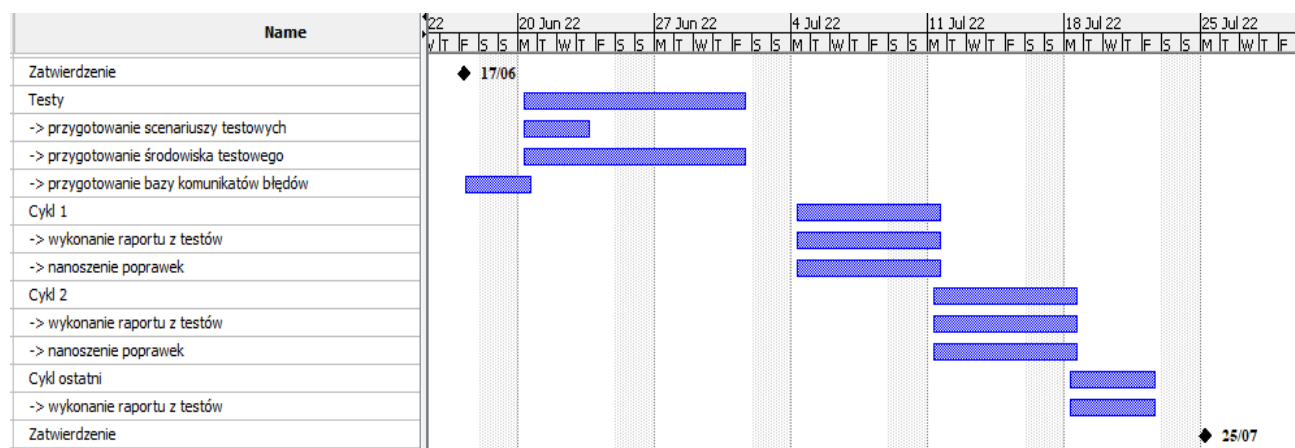
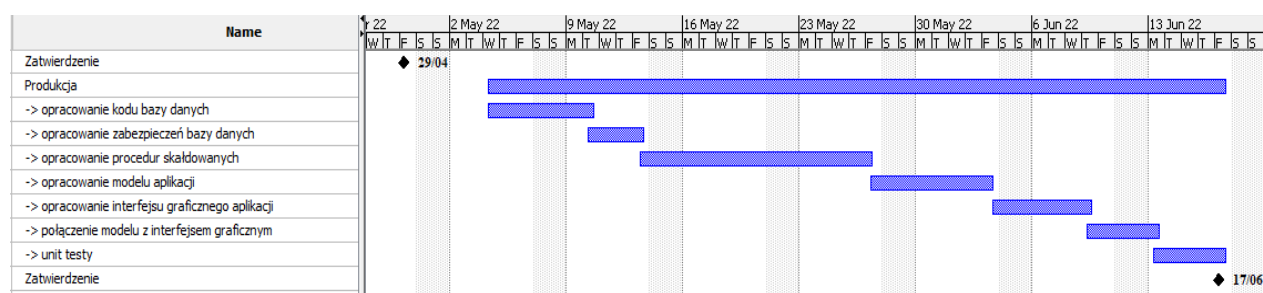
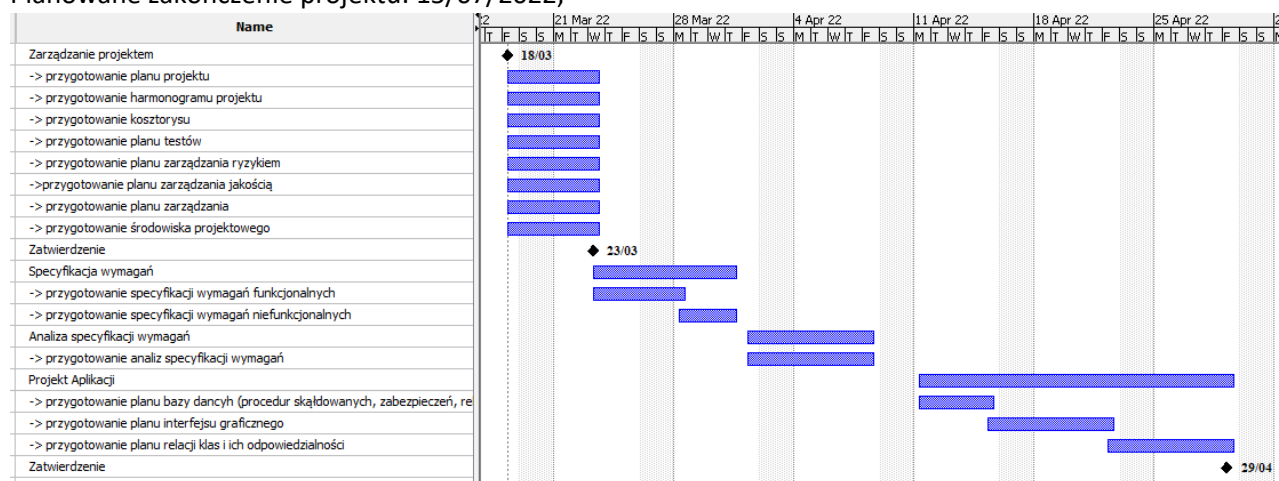
Faza 1 – planowanie od 18/03/2022 do 23/03/2022.

Faza 2 – określania specyfikacji i projektowanie aplikacji od 23/03/2022 do 29/04/2022.

Faza 3 – produkcja od 29/04/2022 do 17/06/2022

Faza 4 – testy (w 3 cyklach testowych) od 17/06/2022 do 25/07/2022

Planowane zakończenie projektu: 15/07/2022;



Więcej szczegółów znajduje się w: *"Harmonogram_projektu_System umawiania wizyt w salonie kosmetycznym.pod"*.

4. Zasoby projektu

Hardware:

[Do pracy przy projekcie]

- 5 komputerów stacjonarnych dla programistów i osób zajmujących się dokumentacją:
 - o minimum 16Gb pamięci ram.
 - o minimum 512Gb pamięci dyskowej SSD.
 - o moc obliczeniowa procesora minimum 2.2GHz.
 - o karta sieciowa: dowolna pozwalająca na komunikację z siecią lokalną.
 - o Co najmniej 4 port USB 3.0

[Lokalne środowisko d testowania systemu]

- Dla urządzeń pozwalających na rejestrację:
 - o minimum 1Gb pamięci ram.
 - o minimum 64Gb pamięci dyskowej SSD.
 - o moc obliczeniowa procesora minimum 1.4-2.0 GHz.
 - o karta sieciowa: dowolna pozwalająca na komunikację lokalną.
 - o Co najmniej 1 post USB 3.0
- Dla urządzenia administratora na którym będzie znajdowała się lokalna baza danych:
 - o minimum 8Gb pamięci ram.
 - o minimum 512Gb pamięci dyskowej SSD.
 - o moc obliczeniowa procesora minimum 2.4-2.7 GHz.
 - o Liczba rdzeni procesora: minimum 2 rdzenie.
 - o karta sieciowa: dowolna pozwalająca na komunikację lokalną.
 - o Co najmniej 3 porty USB 3.0.

Software:

- System operacyjny Windows 10.

5. Standardy i narzędzia w projekcie

Standardy:

- OOP (*Object oriented programming*).
- MVVM (Model View View-Model) - architektura aplikacji.

Narzędzia:

- Visual Studio 2019 – wersja pełna.
- MS_SQL – wersja pełna.
- Microsoft SQL Server Management Studio 18.
- Microsoft Office 365.
- Project-Libre.
- Visual Studio Code – wersja najnowsza.