

Faza konstrukcji

Dominik Budzki, Kacper Kingsford

1. Testy funkcjonalne

a. Test obsługi zamówienia

Inicjator: Kelner

Główny scenariusz:

1. Kelner posiada przygotowane zamówienie klienta
2. Wybiera w aplikacji odpowiedni stolik
3. Dodaje zamówienie do tego stolika
4. Zatwierdza i koszt dolicza się do rachunku
5. Zamówienie wyświetla się na ekranie kucharzom

Krok	Opis	Oczekiwane
1	Podejście do aplikacji aby dodać zamówienie	Na ekranie poprawnie wyświetla się mapa lokalu.
2	Zaznaczanie odpowiedniego stolika	<p>Po kliknięciu na wybrany, zajęty stolik wyświetla się okno modalne z przyciskami "Dodaj zamówienie", "Rachunek", "Zakończ obsługę"...</p> <p>Wybieramy opcję "Dodaj zamówienie".</p>
3	Dodanie odpowiedniego zamówienia	<p>Okno modalne z dwiema kolumnami w postaci "Produkt" - "Ilość". Produkt można wybrać tylko z ustalonej grupy pochodzącej z bazy danych z tabeli "menu". Dostępna jest wyszukiwarka.</p> <p>Przykład: Produkt: "Kotlet schabowy z ziemniakami i surówką", Ilość: 2.</p> <p>Możliwość dodania kolejnego produktu poprzez naciśnięcie przycisku "+" pod ostatnim wierszem. Możliwość dodania notatki do zamówienia</p>

		naciskając przycisk "Dodaj notatkę" usytuowany na dole okna. Na dole przycisk "Zatwierdź" oraz "Anuluj".
4	Zatwierdzenie i dodanie do rachunku	Przycisk "Zatwierdź". Po jego kliknięciu, do bazy danych zostanie wysłana informacja o produkcie zakupionym, ilości oraz cenie. Zapisane to zostaje na rachunku klienta w bazie danych.
5	Zamówienie wyświetlane na aplikacji w kuchni	<p>Nowe zamówienie zostaje pobrane z bazy danych w momencie gdy kelner doda je po swojej stronie. Wyświetla się w postaci "Stolik" - Produkt" - "Ilość". Gdy do zamówienia została dostarczona notatka, ona również wyświetla się pod zamówieniami.</p> <p>Przykład: Stolik: 01, Produkt: "Kotlet schabowy z ziemniakami i surówką", Ilość: 2.</p>

b. Test rezerwacji stolika w restauracji

Inicjator: Klient

Główny scenariusz:

1. Klient wchodzi na stronę internetową restauracji do zakładki z rezerwacją na której znajduje się nasze oprogramowanie
2. Wyświetla się mapa restauracji a na niej widoczne stoliki pokolorowane wg. legendy (zielone - wolne, czerwone - zajęte)
3. Klient zaznacza wybrany stolik. Stolik ten zmienia swój kolor na żółty (wybrany).
4. Klika przycisk "Dalej"
5. Wyświetla się formularz który klient wypełnia odpowiednimi danymi
6. Klient potwierdza i (ewentualnie) otrzymuje potwierdzenie na adres e-mail

Krok	Opis	Oczekiwane
1	Otwarcie strony z aplikacją z widokiem służącym do rezerwacji stolika.	Wyświetla się aktualna mapa lokalu. Wybieramy datę oraz godzinę (format hh:mm dd/mm/rrrr). Stoliki wolne oznaczone są kolorem zielonym, zajęte kolorem czerwonym.
2	Wybranie stolika	Po kliknięciu na wolny stół wyświetla się okno modalne pytające o potwierdzenie wyboru (Tak/Nie). Kliknięcie na zajęty stół nic nie robi. Po potwierdzeniu wyboru, dany stół zostaje tymczasowo zarezerwowany na 5 minut.
3	Uzupełnianie danych	Po zatwierdzeniu wyboru w poprzednim kroku wyświetla się formularz z danymi: Imię, Nazwisko, oraz <u>opcjonalne</u> formy komunikacji: numer telefonu i adres e-mail. Walidacje: - Imię i Nazwisko: Akceptowane wszystkie możliwe imiona i nazwiska w tym. litery specjalne (ą, ę, ü itd.), znaki (d'Arras, Sausage-Hausen) - Numer telefonu: zgodny z api numverify (różne formaty: xxx-xxx-xxx, xxxxxxxxxx, xxxx-xxx-xxx itd.) - adres e- mail: nazwa@domena
4	Zatwierdzanie danych	Dane zostają wysłane do bazy danych i zapisane. Serwer odpowiada czy operacja przebiegła pomyślnie poprzez komunikat "Rezerwacja zakończona pomyślnie!".
5a	Faza końcowa	W przypadku udanej operacji użytkownik zostaje poinformowany o tym tekstem o pomyślnej rezerwacji miejsca. Gdy klient wypełnił pole z

		adresem e-mail, na jego konto zostaje wysłany wygenerowany e-mail potwierdzający rezerwację na konkretną datę i godzinę.
5b	(Ewentualne) niepowodzenie operacji	Użytkownik zostaje powiadomiony tekstem o nieudanej operacji rezerwacji tekstem "Operacja zakończona niepowodzeniem!", zostaje poproszony o powtórzenie operacji.

c. Test edycji mapy lokalu

Inicjator: Pracownik

Główny scenariusz:

1. Pracownik posiada włączoną aplikację z widokiem mapy lokalu
2. Klika przycisk "Edytuj mapę"
3. Ekran zmienia się w tryb edycji: Stoliki posiadają jednolity kolor (np. czarny) oraz po naciśnięciu na nie, można je przesunąć, obracać przyciskami lub ręcznie.
4. Po uzyskaniu zadowalającego wyniku klika "Zakończ edycję". Mapa zostaje zapisana i od teraz można operować na tej nowej mapie

Krok	Opis	Oczekiwane
1	Podejście do aplikacji, do ekranu mapy lokalu	Na ekranie poprawnie wyświetla się aktualna mapa lokalu.
2	Rozpoczęcie edycji mapy	Kliknięcie przycisku "Edytuj mapę". Stoliki zmieniają swoje zachowanie: po ich kliknięciu można je przesunąć, obracać i edytować ich rozmiar. Są pokolorowane na czarno, po kliknięciu zmieniają kolor na biały.
3	Edycja mapy.	Możliwość przesuwania stolików poprzez najechanie kursorem i przytrzymanie go na stoliku. Możliwość zmiany ich kształtu,

		rozmiaru. Po kliknięciu można skorzystać z przycisków z boku ekranu do przesuwania i obracania.
4	Koniec edycji	Po uzyskaniu dowolnego układu naciśnięcie przycisku "Zakończ edycję" spowoduje zapisanie danego układu. Możliwość zapisania kilku układów pod różnymi nazwami Format nazwy: litery [a-z][A-Z], długość 26 liter.

2. Pomiar dla wymagań niefunkcjonalnych (ISO/IEC-9126)

a. Kompletność funkcjonalności

Pomiar opisany wzorem $X = 1 - A/B$, gdzie A - liczba brakujących funkcji, B - liczba zaimplementowanych funkcji.

Cel: $X = 1$

b. Efektywność działania, czas reakcji na zapytania

Aplikacja na średniej klasy sprzęcie powinna obsługiwać dowolne zapytanie do bazy oraz wyświetlać ich wyniki w maksymalnie 1 sekundę. Nie operujemy na wielkich danych, więc taki limit wydaje się sensowny.

c. Użyteczność, łatwość korzystania z aplikacji

Pomiar opisany wzorem $X = N/T$, gdzie T - czas działania w minutach, N - liczba błędów użytkownika (złe zrozumienie obsługi, proste pomyłki, dłuższe zastanowienie się nad kolejnym krokiem). Cel: $X = 0$

d. Niezawodność

Pomiar opisany wzorem $X = A/T$, gdzie A - liczba błędów aplikacji, T - czas używania aplikacji w minutach. Cel: $X < 0.01$.

3. Plan beta testowania

Testy beta podzielone na trzy fazy. W drugiej i trzeciej fazie będziemy musieli znaleźć restauracje, które zgodzą się na przetestowanie naszego systemu. W ramach zachęty, w przypadku pomyślnego okresu testowego zgodzimy się by dane restauracje korzystały z naszej aplikacji bezpłatnie na okres 6 miesięcy.

1. Pierwsza faza testowania

W tej fazie nasz tester zajmie się testowaniem funkcjonalności na podstawie planów testowania. Na tym etapie sam będzie musiał zadbać o poprawność danych. Ważne jest to aby na tym etapie podstawowe funkcjonalności działały poprawnie

2. Druga faza testowania

W tej fazie skorzystamy z uprzejmości jednej restauracji, z którą wcześniej się umówiliśmy. Przeanalizujemy aplikację pod kątem wymagań funkcjonalnych jak i niefunkcjonalnych np. Ile czasu zajmie pracownikom nabranie biegłości w obsłudze oprogramowania. Będziemy również w stanie sprawdzić działanie aplikacji na realnych danych.

3. Trzecia faza testowania

Produkt zostanie udostępniony małej (np. 5) grupie restauracji chętnych do sprawdzenia naszej aplikacji. W tym etapie sprawdzimy jak oprogramowanie sprawdza się w różnych środowiskach funkcjonujących jednocześnie.

4. Plan zarządzania jakością

- Testowanie jednostkowe na bieżąco funkcji napisanych przez programistów
- Testowanie bazy oraz aplikacji pod kątem bezpieczeństwa
- Aplikacja będzie pisana zgodnie z podstawowymi standardami kodowania
- Programiści pod koniec każdego tygodnia raportują swoje postępy
- Praca będzie kontrolowana w systemie kontroli wersji Git.
- Na koniec zostanie sporządzona dokumentacja.

5. Plan wykonania produktu

Nr	Opis	Czas	Początek	Koniec
1	Zebranie potrzebnych informacji	3 dni	01.02.2022	03.02.2022
2	Sporządzenie specyfikacji wymagań	3 dni	04.02.2022	08.02.2022
3	Przygotowanie architektury systemu	5 dni	09.02.2022	15.02.2022
4	Projekty interfejsu	4 dni	16.02.2022	21.02.2022
5	Organizacja pracy	2 dni	22.02.2022	24.02.2022
6	Programowanie	45 dni	25.02.2022	17.04.2022
7	Testy jednostkowe	7 dni	18.04.2022	26.04.2022
8	Wprowadzanie poprawek po testach	5 dni	27.04.2022	05.05.2022
9	Dalsza implementacja	30 dni	06.05.2022	16.06.2022
10	Pierwsza faza beta testowania	7 dni	17.06.2022	27.06.2022
11	Testowanie bezpieczeństwa i jakości	10 dni	28.06.2022	11.07.2022
12	Poprawki błędów	7 dni	12.07.2022	20.07.2022
13	Wdrażanie systemu	10 dni	21.07.2022	03.08.2022
14	Druga faza beta testowania	7 dni	04.08.2022	12.08.2022
15	Poprawki, ulepszenia na podstawie testów	5 dni	16.08.2022	22.08.2022
16	Trzecia faza testowania	10 dni	23.08.2022	06.09.2022
17	Ostatnie poprawki błędów	7 dni	07.09.2022	15.09.2022
18	Stworzenie dokumentacji	7 dni	16.09.2022	26.09.2022
19	Wypuszczenie produktu	-	27.09.2022	

*do ilości dni nie liczą się weekendy oraz dni ustawowo wolne.

6. Ocena zgodności wykonanych prac ze specyfikacją wymagań

Dokonaliśmy pewnych poprawek w bazie danych. Tabela "Potrawy" powinna nazywać się "Menu".

Do tego okres testowania będzie dłuższy niż zakładaliśmy podczas tworzenia tablicy koncepcyjnej, wtedy założyliśmy łącznie 3 tygodnie na testowanie. Z planu wykonania produktu wynika że będziemy potrzebować około 5/6 tygodni. By przyspieszyć okres testowania moglibyśmy zatrudnić drugiego testera. To może być dobra inwestycja w kontekście czasowym, jak i jakościowym, ponieważ uda nam się wtedy wykryć więcej błędów i problemów.