

# Recenzja od łososiowych

## części 2 projektu

### Błędy opracowania

- Scenorys jest trochę słabo widoczny.
- Warto zdefiniować co to jest RCP, ponieważ nie wiadomo co to jest.
- W zdaniu “Z wyborem form kontaktu również poradziła sobie dobrze – w karcie kontakty zaznaczyła odpowiednie pola wyboru.” słowo “kontakty” powinno być w cudzysłowie.
- “Osoba poproszona o skorzystanie z naszego interfejsu **weszła w zakładkę „Pracownicy”** następnie miała drobny problem ze **znalezieniem przycisku odpowiedzialnego za dodanie nowego pracownika**, jednak po około 10 sekundach znalazła go i bezbłędnie wypełniła formularz, kończąc zadanie. ” - W tym zdaniu złożonym brakuje przecinka lub spójnika między jednym czasownikiem (“weszła”) a drugim (“miała”).
- “Użytkownik stworzył grafik, zmieniając kartę na „Grafik” **a** następnie wybierając opcję „Utwórz nowy”. ” - W tym zdaniu brakuje przecinka.

Poza wyżej wymienionymi rzeczami, praca jest dobrze zreferowana.

### Błędy interfejsów

Obydwa interfejsy nie są najlepsze. Patrząc na nie, czujemy się zmieszani i nie do końca wiemy, do czego to wszystko tak naprawdę służy. Mamy wrażenie ogólnego braku hierarchii i czujemy, że ciężko jest się nauczyć używać tej aplikacji. Ikony są zdecydowanie za duże, a nagłówki się gubią w tle. Wszelkie kafelki na grafiku wyglądają identycznie jak przyciski, co jest bardzo mylące. Za mało wolnych przestrzeni, wszystko jest zbyt ściśnięte, przez co różne elementy interfejsu na siebie nachodzą. Panuje ogólny chaos. Będąc na jakiejś podstronie, nie wiemy, na jakiej konkretnej tak naprawdę jesteśmy, ponieważ ani menu, ani nagłówki tego nie sygnalizują (w prototypie 2 jest już lepiej).

Warto to poprawić jeśli planujecie dalej rozwijać któryś z tych prototypów. Naszym zdaniem prototyp 2 jest lepszy.

## **Analiza według 10 heurystyk Jakoba Nielsena**

(-) koło nazwy oznacza, że naszym zdaniem interfejs nie spełnia danej heurystyki, a (+) oznacza, że spełnia.

### **Prototyp 1:**

#### **1. Pokazuj status systemu (-)**

Będąc na danej podstronie, nie wiemy, gdzie jesteśmy. Brak nagłówków bądź innej sygnalizacji. Do tego przyciski wyglądają jak nieklikalne elementy w kalendarzu, co bardzo myli i nie sygnalizuje, co się dzieje. Czyli brak sygnalizacji o statusie aplikacji.

#### **2. Zachowaj zgodność pomiędzy systemem a rzeczywistością (-)**

Interfejs wygląda inaczej niż inne dobrze znane rozwiązania. Jest raczej nieznany użytkownikowi. Do tego jest nieintuicyjny, przez co nie można powiedzieć, że "mówi językiem użytkownika".

#### **3. Daj użytkownikowi pełną kontrolę (+)**

Tutaj interfejs sprawuje się całkiem dobrze. Użytkownik może dużo zrobić samemu (podjąć dużo akcji) i zawsze ma możliwość wyjścia z danej akcji. Używając tej platformy, czujemy się u władzy, czujemy, że to ja kontroluję tę aplikację.

#### **4. Trzymaj się standardów i zachowaj spójność (+)**

Platforma zachowuje spójność interfejsu.

#### **5. Zapobiegaj błędom (-)**

Naszym zdaniem interfejs nie zapobiega błędom wystarczająco dobrze. Na przykład przyciski wyglądają jak nieklikalne elementy w

kalendharzu, co może spowodować całą gamę potencjalnych błędów i niedogodności.

**6. Pozwalaj wybierać, zamiast zmuszać do pamiętania (-)**

Interfejs jest nieintuicyjny, przez co trzeba zapamiętać, jak się go używa. Niestety nie wskazuje on sam co i jak można zrobić.

**7. Zapewnij elastyczność i efektywność (-)**

Użytkownik początkujący będzie bardzo niewydajny, natomiast użytkownik doświadczony może być wydajny. Mimo wszystko, początkujący mogą mieć problem.

**8. Dbaj o estetykę i umiar (-)**

Interfejs zawiera wyłącznie rzeczy potrzebne, natomiast nie jest estetyczny. Przez duże natężenie elementów na małej przestrzeni, ciężko jest znaleźć to, czego się szuka.

**9. Zapewnij skuteczną obsługę błędów (-)**

Brak informacji zwrotnych więc nie da się tego określić.

**10. Zadbaj o pomoc i dokumentację (-)**

Dokumentacja dołączona do pliku jest uboga. Brak jakichkolwiek pomocy w uczeniu się obsługi aplikacji.

- **Wynik: 2/10**

## **Prototyp 1:**

**1. Pokazuj status systemu (+)**

Nagłówki sygnalizują, gdzie się znajdujemy.

**2. Zachowaj zgodność pomiędzy systemem a rzeczywistością (-)**

Interfejs wygląda inaczej niż inne dobrze znane rozwiązania. Jest raczej nieznanym użytkownikowi. Do tego jest dość nieintuicyjny, przez co nie można powiedzieć, że “mówi językiem użytkownika”.

**3. Daj użytkownikowi pełną kontrolę (+)**

Tutaj interfejs sprawuje się całkiem dobrze. Użytkownik może dużo zrobić samemu (podjąć dużo akcji) i zawsze ma możliwość wyjścia z danej akcji. Używając tej platformy, czuję się u władzy, czuję, że to ja kontroluję tę aplikację.

**4. Trzymaj się standardów i zachowaj spójność (+)**

Platforma zachowuje spójność interfejsu.

**5. Zapobiegaj błędom (+)**

W tym zakresie interfejs sprawuje się dobrze.

**6. Pozwalaj wybierać, zamiast zmuszać do pamiętania (-)**

Interfejs jest dość nieintuicyjny przez co, trzeba zapamiętać, jak się go używa. Niestety często nie wskazuje on sam co i jak można zrobić.

**7. Zapewnij elastyczność i efektywność (-)**

Użytkownik początkujący będzie bardzo niewydajny, natomiast użytkownik doświadczony może być wydajny. Mimo wszystko, początkujący mogą mieć problem.

**8. Dbaj o estetykę i umiar (+)**

Interfejs zawiera wyłącznie rzeczy potrzebne i jest całkiem estetyczny. Mimo dość dużego natężenia elementów na małej przestrzeni, nie jest ciężko znaleźć, czego się szuka.

**9. Zapewnij skuteczną obsługę błędów (-)**

Brak informacji zwrotnych więc nie da się tego określić.

**10. Zadbaj o pomoc i dokumentację (-)**

Dokumentacja dołączona do pliku jest uboga. Brak jakichkolwiek pomocy w uczeniu się obsługi aplikacji.

- **Wynik: 5/10**