

## Recenzja dokumentacji zespołu nr 3

### 1. Określenie, czy wszystkie wymagane elementy dokumentacji są w niej zawarte

Brakuje opisu diagramu klas oraz po jednym diagramie z każdego rodzaju (aktywności, stanu).

### 2. Opisanie każdego elementu dokumentacji pod względem poprawności z istniejącymi standardami

- *diagram przypadków użycia*: poprawnie;
- *opis diagramu przypadków użycia*: poprawnie;
- *wymagania нефункционалне*: niezgodny ze standardem ustalonym przez prowadzącego (opisowo zamiast tabelarycznie);
- *wymagania funkcjonalne*: w większości poprawnie (brak opisu poszczególnych funkcji);
- *diagram aktywności*: niepoprawnie: błędne użycie bramek i opcji decyzji;
- *diagram klas*: nie do końca poprawnie; pole nie może się nazywać „class” (klasa Booking, Airplane), klasy połączone ze sobą mają takie same nazwy pól;
- *diagram pakietów*: poprawnie;
- *diagram stanów*: poprawnie;
- *baza danych*: niepoprawnie (nieprawidłowe nazwy kolumn i kluczy (powtarzają się), brak konkretnej notacji);
- *opis bazy danych*: nie do końca zgodny ze standardem ustalonym przez prowadzącego.

### 3. Określenie przejrzystości każdego elementu oraz jego zgodności z pozostałymi elementami dokumentacji

- *diagram przypadków użycia*: mogłyby zostać wprowadzone drobne poprawki w celu poprawienia przejrzystości diagramu;
- *opis diagramu przypadków użycia*: przejrzyste; brak opisu czterech przypadków użycia: dodaj bilet, anulowanie kupna biletu, dodaj punkty, sprawdzanie harmonogramu pracy, dużo błędów stylistycznych;
- *wymagania нефункционалне*: przejrzyste; zbyt ogólne (wielojęzyczność – jakie języki konkretnie?);
- *wymagania funkcjonalne*: przejrzyste; niektóre przypadki użycia powstały najprawdopodobniej intuicyjnie i nie zostały zawarte w wymaganiach funkcjonalnych;
- *diagram aktywności*: przejrzysty;
- *diagram klas*: bardzo nieprzejrzysty, brak konsekwencji w formatowaniu, nie wiadomo, o co chodzi;
- *diagram pakietów*: przejrzysty;
- *diagram stanów*: przejrzysty;
- *baza danych*: niedbałe i niedokładne połączenia między tabelami, brak konkretnej notacji utrudnia odczytanie;
- *opis bazy danych*: nieprzejrzysty, niedbały.

### 4. Określenie, czy dokumentacja zgadza się z wymaganiami specyfikacji

- *diagram przypadków użycia*: pracownicy nie mają możliwości sprawdzenia informacji na temat poszczególnych lotów, a powinni mieć;
- *opis diagramu przypadków użycia*: w związku z tym, że brakuje powyższego przypadku w diagramie, nie jest on opisany;

- *wymagania нефunkcjonalne*: poprawnie – specyfikacja nie uwzględnia wytycznych na temat wymagań нефunkcjonalnych;
- *wymagania funkcjonalne*: brakuje wymagania o ustalaniu cen za przeloty, odnośnie odsprzedaży biletów, wyświetlania informacji na temat poszczególnych lotów przez pracowników;
- *diagram aktywności*: zgodny;
- *diagram klas*: zgodny;
- *diagram pakietów*: zgodny;
- *diagram stanów*: zgodny;
- *baza danych*: zgodny;
- *opis bazy danych*: zgodny.

## 5. Opinia grupy na temat poszczególnych elementów, propozycja poprawek

- *diagram przypadków użycia*: niepotrzebny aktor „Klient” (nie wnosi nic do diagramu), niepoprawne użycie parametru „include” pomiędzy przypadkami „Zakup biletu” a „Odesłanie biletu”;
- *opis diagramu przypadków użycia*: niepoprawny opis przypadku „Rejestracja” (żeby się zarejestrować trzeba się zalogować?), „Utworzenie grafiku dla pilotów i stewardess” (brak konsekwencji, w tym jednym przypadku w momencie niepowodzenia przypadek się kończy, w innych następuje próba ponownego wykonania przypadku), „Logowanie do systemu” (cel: ustalenie nagród?, niepoprawne zakończenie), dużo błędów stylistycznych;
- *wymagania нефunkcjonalne*: spora część wymagań нефunkcjonalnych jest niepoprawnie zidentyfikowanych (od 8. punktu włącznie), imię i nazwisko pilota nie powinna być do wglądu dla zwykłego użytkownika;
- *wymagania funkcjonalne*: jest ich za mało jak do tak rozbudowanego systemu;
- *diagram aktywności*: brak propozycji poprawek;
- *diagram klas*: grupa zaproponowała ponowne utworzenie diagramu klas, lub wprowadzenie znaczących zmian w istniejącym ze względu na brak opisu oraz przeznaczenia poszczególnych klas;
- *diagram pakietów*: brak poprawek;
- *diagram stanów*: brak poprawek;
- *baza danych*: największym błędem zauważonym przez grupę jest brak konsekwencji podczas stosowania nazewnictwa atrybutów, brak jednoznacznej notacji; zaproponowano ustalenie jednej notacji i zmiany nazw atrybutów;
- *opis bazy danych*: brak poprawek.

## 6. Podsumowanie

Ogólnie projekty dobre, jednak ciężko zrozumieć diagram klas oraz ERD. Zawiera drobne błędy, które będą na bieżąco korygowane podczas implementacji.