
System zarządzania hotelem

Programowanie Zdarzeniowe

Łukasz Świtaj

semestr 18L

Spis treści:

1. Założenia oraz zarys projektu.	
a. Wstępne założenia projektu	2
b. Model implementacji	
i. Opis użytego wzorca projektowego	3
ii. Opis klas w odniesieniu do wzorca	3
c. Użyte technologie	4
2. Funkcjonalności. Założenia użytkowe z uwzględnieniem ról	5
a. Wspólne	
b. Szef/manager	
c. Pracownik	
d. Klient	

Założenia oraz zarys projektu

Wstępne założenia projektu

System zarządzania miał zostać stworzony na potrzeby firmy mającej jeden hotel, aby zautomatyzować oraz zabezpieczyć takie codzienne czynności jak:

- składanie rezerwacji przez klientów
- edytowanie rezerwacji online przez pracowników
- kontrolowanie stanu zapelnienia hotelu w poszczególnych okresach
- nadawanie awansów pracownikom
- kontrolowanie statystyk dotyczących przychodów oraz wydatków hotelu

Powyższe zostało zrealizowane dzięki implementacji produktu informatycznego mającego podane funkcjonalności.

Model implementacji

Opis użytego wzorca projektowego

Projekt został zrealizowany w formie serwera zarządzanego przez hotel.

Całość implementacji została napisana we wzorcu Model View Controller, w którym każdy z modułów stanowi odrębną część projektu dzięki czemu jest on niewrażliwy na zmiany pozostałych.

Opis klas w odniesieniu do wzorca

W celu zachowania rozdzielności między poszczególnymi warstwami aplikacji zostały utworzone pakiety skupiające się na każdej warstwie z osobna.

Każda z klas skupia się na edycji/wywoływaniu metod związanych ze swoim odpowiednikiem mini-świata przyjętego w projekcie (np. klasy Employee* pobierają użytkowników z bazy, sumują ich pensje itd.). Wyjątek stanowi kontroler – dla ułatwienia zarządzania kodem są tam przyporządkowane funkcjonalności „per uprawnienie” (przykładowo – jeśli tylko pracownik jest upoważniony do wprowadzania zmian w rezerwacjach to właśnie w kontrolerze pracownika będzie funkcja odpowiedzialna za reakcję na mapowanie /zmieńRezerwację).

Pakiety, które odpowiadają za poszczególne warstwy modułu MVC to kolejno:

- paczka .config – konfiguracja kontekstu aplikacji oraz modułów odpowiedzialnych m. in. za bezpieczeństwo
- paczka .controller – kontroler odpowiedzialny za mapowanie adresów www na poszczególne widoki oraz przyporządkowywanie im metod z warstwy Service (model). Kontroler w
- paczka .domain – deklaracje klas oraz implementacje ich metod
- paczka .repository – model lub inaczej warstwa danych odpowiedzialna za obsługę połączenia z bazą danych
- paczka .service – model odpowiedzialny za przetworzenie danych dostarczonych przez warstwę danych i kontakt z kontrolerem
- folder /src/main/webapp/WEB-INF/views – warstwa

Użyte technologie

System został utworzony przy użyciu Springa MVC oraz środowiska Spring Tool Suite. Dodatkowe technologie użyte do realizacji projektu:

- Hibernate – do połączenia z bazą danych
- Hibernate Validator – do walidacji formularzy
- Spring Security – do zabezpieczenia haseł, nadawania uprawnień oraz logowania
- Spring Security – do zabezpieczenia haseł, nadawania uprawnień oraz logowania

Funkcjonalności

Założenia użytkowe z uwzględnieniem ról

Każdy użytkownik może:

- zalogować się przez stronę /login, dzięki czemu zostaje przekierowany na właściwą dla siebie stronę główną.

Każdy manager może:

- dodać nowego pracownika po wybraniu opcji „Add New Employee”
- dowiedzieć się, jaki jest stosunek wydatków firmy do przychodów „Show statistics”
- zmienić stanowisko oraz wypłatę pracownika „Promote Employee”

Każdy pracownik może:

- wpisać login klienta, po jego wcześniejszej prośbie, który przeniesie go do formularza umożliwiającego zmianę wybranej rezerwacji

Każdy klient może:

- obejrzeć dotychczas dokonane rezerwację przez domyślną stronę /myReservations
- dokonywać rezerwacji przez stronę /booking