



POLITECHNIKA ŚLĄSKA
WYDZIAŁ AUTOMATYKI, ELEKTRONIKI I
INFORMATYKI
KIERUNEK INFORMATYKA

Projekt inżynierski

Projekt i implementacja aplikacji internetowej do obsługi
małego hotelu

Autor: Jakub Łagódka

Kierujący pracą: dr hab. prof. Pol. Śl. Paweł Kasprowski

Gliwice, Styczeń 2021

Oświadczenie

Wyrażam zgodę/nie wyrażam* zgody na udostępnienie mojej pracy
dyplomowej/rozprawy doktorskiej*

....., dnia

.....
(podpis)

.....
(poświadczenie wiarygodności podpisu przez Dziekanat)

* właściwe podkreślić

Spis treści

1.	Wstęp.....	7
1.1.	Cel i zakres pracy	7
1.2.	Plan pracy.....	8
2.	Analiza tematu	9
2.1.	Narodziny hotelarstwa	9
2.2.	Czym właściwie jest hotelarstwo oraz usługa hotelarska	9
2.3.	Rodzaje obiektów hotelarskich	10
2.4.	Kategoryzacja – liczba gwiazdek	11
2.5.	Klasyfikacja usług hotelarskich.....	12
2.6.	Doba hotelowa.	13
2.7.	Organizacja hotelu – recepcjoniści.	13
2.8.	Rezerwacje	14
2.9.	Pokoje noclegowe.....	16
2.10.	Systemy do obsługi hotelu	17
3.	Wymagania i narzędzia	19
3.1.	Wymagania funkcjonalne	19
3.2.	Wymagania нефункционалне.....	19
3.3.	Przypadek użycia – diagram UML.....	20
3.4.	Opis narzędzi użytych w programie	219
3.5.	Opis pracy nad projektowaniem i implementacją systemu.....	23
4.	Specyfikacja zewnętrzna	25
4.1.	Wymagania sprzętowe oraz programowe systemu	25
4.2.	Sposób instalacji i aktywacji.....	25
4.3.	Kategorie użytkowników dostępnych w systemie	25
4.4.	Sposób obsługi programu	25
4.5.	Administracja systemem.....	25
4.6.	Kwestie bezpieczeństwa	25

4.7.	Scenariusze korzystania z systemu - przykład działania aplikacji	25
5.	Specyfikacja wewnętrzna.....	37
5.1.	Idea działania aplikacji.....	37
5.2.	Architektura systemu	37
5.3.	Opis użytych struktur danych, a także organizacji bazy danych ...	37
5.4.	Opis wykorzystanych komponentów, modułów, bibliotek, ważniejszych klas użytych w programie	37
5.5.	Szczegółowy opis implementacji wybranych fragmentów programu.....	37
6.	Weryfikacja i walidacja	45
6.1.	Sposób testowania w ramach pracy.....	45
6.2.	Organizacja eksperymentów.....	45
6.3.	Wykryte i usunięte błędy.	46
7.	Podsumowanie i wnioski	47
	Bibliografia	i
	Spis skrótów i symboli	iv
	Zawartość dołączonej płyty	vi
	Spis rysunków	vii
	Spis tabel.....	ix

1. Wstęp

Korzystanie z usług noclegowych stało się niezwykle popularne. Dla przykładu w przypadku wyjazdu na wakacje jednym z pierwszych wyzwań stojących przed wczasowiczem jest znalezienie oraz rezerwacja zakwaterowania. Z kolei jednym z największych utrudnień jest uzyskanie informacji na temat dostępności wolnych pokoi, gdyż większość małych hoteli czy pensjonatów, w przypadku gdy posiada własną stronę internetową, takowa strona jest mało funkcjonalna i nie umożliwia dostępu do aktualnych danych, takich jak obłożenie – czy w danym hotelu znajdują się dostępne w konkretnym terminie pokoje do wynajęcia. W takim wypadku potencjalny klient jest zmuszony do skontaktowania się z wynajmującym, zwykle na stronie znajduje się numer telefonu do takiej osoby, w celu zorientowania się, czy wynajmujący posiada wolne pokoje. Oczywiste jest, że wykonywanie połączeń telefonicznych do wielu wynajmujących nie jest najwygodniejszym rozwiązaniem dla potencjalnego klienta, podobnie sprawa tyczy się samych wynajmujących. Z takich pobudek właśnie w tej pracy przedstawiono rozwiązanie takiego problemu.

1.1 Cel i zakres pracy

Została stworzona aplikacja internetowa umożliwiająca rezerwację pokoi poprzez stronę internetową, a także anulowanie rezerwacji poprzez klienta. Recepcjonista ma możliwość zobaczenia podglądu obłożenia. Produkt jest

przeznaczony dla małego zakładu hotelarskiego, który jest zainteresowany podniesieniem jakości obsługi klienta oraz zwiększeniem produktywności pracy przez pracowników obsługi.

1.2 Plan pracy

W rozdziale drugim opisane zostały zagadnienia związane z działalnością usług hotelarskich, a także przedstawiono inne internetowe systemy zarządzana hotelem. Rozdział trzeci opisuje wymagania funkcjonalne oraz niefunkcjonalne aplikacji, zostały w nim również opisane narzędzia użyte w trakcie implementacji systemu. Rozdział czwarty opisuje specyfikację zewnętrzną programu, podane w nim zostały wymagania sprzętowe i programowe, sposób instalacji systemu, kategorie użytkowników istniejących w systemie, sposób obsługi oraz administrowania aplikacją. Poruszone w nim zostały także kwestie bezpieczeństwa oraz przedstawiony został przykład działania systemu wraz ze zrzutami ekranu. W rozdziale piątym opisana została specyfikacja wewnętrzna aplikacji: przybliżono architekturę aplikacji, opisano struktury danych oraz organizację bazy danych, a także szczegółowo przedstawiono fragment programu wraz z listingiem kodu źródłowego. Rozdział szósty zawiera omówienie procesu testowania systemu. Praca kończy się na rozdziale siódmym zawierającym podsumowanie, a także możliwe dalsze kierunki rozwoju systemu.

2. Analiza tematu

W rozdziale tym dokonano analizy informacji zawartych w kilku różnych źródłach. Wykorzystano przy tym różne dostępne źródła.

2.1 Narodziny hotelarstwa

Już od tysięcy lat istniała potrzeba podróżowania oraz zmiany miejsca swojego pobytu, tak więc hotelarstwo narodziło się już w bardzo odległych czasach. Istnieje możliwość zapoznania się z dokładniejszym opisem historii hotelarstwa przedstawionego w artykule internetowym, adres internetowy strony znajduje się w bibliografii [4].

2.2 Czym właściwie jest hotelarstwo oraz usługa hotelarska

Zgodnie z definicją hotelarstwo jest działalnością gospodarczą prowadzącą się do udzielania gościny osobom przyjezdnym w przeznaczonych do tego miejscach noclegowych. Dokładniejsza analiza hotelarstwa znajduje się w załączonym w bibliografii odnośniku internetowym [5]. W celu precyzyjnej, oficjalnej odpowiedzi na pytanie czym

jest usługa hotelarska warto sięgnąć do ustawy z dnia 29 sierpnia 1997 r. o usługach hotelarskich oraz usługach pilotów wycieczek i przewodników turystycznych – odnośnik do ustawy znajduje się w bibliografii [6]. W tejże ustawie znajduje się regulacja prawna niektórych usług hotelarskich.

2.3 Rodzaje obiektów hotelarskich

Możliwe jest dokonanie podziału obiektów hotelarskich na rodzaje, zgodnie z ich zakresem działalności. Na podstawie artykułu 36 przytaczanej już w poprzednim podrozdziale ustawy o usługach hotelarskich oraz usługach pilotów wycieczek i przewodników turystycznych [6] zostały określone definicje tychże obiektów:

- a) Hotele są obiektami posiadającymi minimum dziesięć pokoi, z których to pokoi większość powinno być jedno lub dwuosobowa. Hotele powinny świadczyć kompleksowe usługi związane z pobytem w nich klientów.
- b) Motele są hotelami dodatkowo znajdującymi się przy drogach, posiadające do dyspozycji klienta parking oraz umożliwiającymi korzystanie z usług motoryzacyjnych.
- c) Pensjonaty są to obiekty rozporządzające co najmniej siedmioma pokojami, powinny one świadczyć dla klientów całodienne wyżywienie.
- d) Kempingi to obiekty strzeżone, dające możliwość noclegu przyczepach samochodowych, namiotach, domkach turystycznych czy też innych obiektach stałych, a także przyrządzania posiłków oraz parkowania samochodów.
- e) Domy wycieczkowe są to obiekty dysponujące minimum trzydziestoma miejscami noclegowymi, przystosowanymi do

samoobsługi klientów, a także świadczące co najmniej minimalnym zakresem usług związanych z pobytem w nich klientów.

- f) Schroniska młodzieżowe są obiektami, które są przeznaczone do turystyki młodzieżowej, zarówno indywidualnej, jak i grupowej oraz są dostosowane do samoobsługi przez klientów.
- g) Schroniska to obiekty znajdujące się poza obszarami zabudowanymi, znajdujące się przy szlakach turystycznych, powinny one świadczyć usługi związane z pobytem w nich klientów w co najmniej minimalnym zakresie.
- h) Pola biwakowe są obiektami niestrzeżonymi, umożliwiającymi nocleg w namiotach.

Niektóre obiekty nie zostały zdefiniowane w tej ustawie, jak chociażby ośrodki wczasowe, domy studenckie, internaty czy kwatery prywatne.

2.4 Kategoryzacja – liczba gwiazdek

Baza noclegowa zgodnie z rozporządzeniem ministra gospodarki i pracy z dnia 19 sierpnia 2004 roku, w sprawie obiektów hotelarskich i innych obiektów, w których świadczone są usługi hotelarskie [7], została podzielona na następujące kategorie:

- a) Hotele, motele oraz pensjonaty – pięć kategorii oznaczanych za pomocą gwiazdek.
- b) Kempingi – cztery kategorie oznaczane za pomocą gwiazdek.
- c) Schroniska młodzieżowe oraz domy wycieczkowe – trzy kategorie oznaczane za pomocą cyfr rzymskich.

Przywołane tutaj rozporządzenie określa również wymagania konieczne do spełnienia przez poszczególne kategorie. Każda z nich posiada ściśle zdefiniowane kryteria konieczne do spełnienia przez dany obiekt, określające minimalny poziom świadczonych tamże usług. Dzięki takiemu podziałowi możliwa jest standaryzacja obiektów hotelarskich. Bardzo dobrze znana jest pięciogwiazdkowa kategoryzacja hoteli, powszechna jest wiedza o wzroście standardu hotelu wraz ze zwiększaniem się liczby gwiazdek. Jednakże nie zawsze sprawa jest tak prosta, na jaką wygląda, albowiem w innych krajach europejskich kategoryzacja może się różnić od przedstawionej w tej pracy kategoryzacji obowiązującej na terenie Polski, poprzez obowiązywanie tam innych wymagań dla danej kategorii. Z kolei w niektórych krajach, chociażby w Niemczech, kategoryzacja nie jest uregulowana prawnie, kategorie obiektów są dowolnie używane przez biura podróży. Warto zdawać sobie z tego sprawę. W celu bardziej dogłębnego zapoznania się z tematem kategoryzacji hoteli, wskazane jest zapoznanie się ze stroną internetową, do której odnośnik znajduje się w bibliografii [8]

2.5 Klasyfikacja usług hotelarskich

Współcześnie hotele świadczą szeroki zakres usług, jednakże podstawowymi usługami są usługi noclegowe oraz gastronomiczne. Z kolei usługi nie mieszczące się w definicji usług podstawowych, dzieli się na:

- a) Usługi uzupełniające – związane bezpośrednio z usługami podstawowymi, będącymi ich dopełnieniem. Klient jest zmuszony do korzystania z nich w trakcie pobytu. Zazwyczaj nie wymagają one żadnej dodatkowej opłaty.
- b) Usługi fakultatywne – mają za zadanie uatrakcyjnić pobyt w hotelu. Mogą być płatne.

- c) Usługi towarzyszące – są to dodatkowe usługi oferowane przez hotel, w ogóle nie związane z usługami podstawowymi. W celu korzystania z nich klient nie musi być hotelowym gościem. Wymagają dodatkowej opłaty.

Bardziej szczegółowy opis podziału usług hotelarskich można znaleźć na stronie internetowej, do której odnośnik załączono w bibliografii [9].

2.6 Doba hotelowa

Warto pamiętać o tym, iż w hotelu obowiązuje doba hotelowa, różna od zwykłej doby. Zgodnie z definicją doba hotelowa jest czasem, podczas którego gość ma możliwość przebywania w danym pokoju hotelowym. Zazwyczaj kończy się ona o godzinie dwunastej, tak więc przed tą godziną należy oddać klucze do pokoju hotelowego. Z kolei godzina rozpoczęcia doby hotelowej jest mniej oczywista, często jest to godzina czternasta, jednakże w zależności od aktualnego obłożenia, istnieje czasem możliwość udostępnienia pokoju klientowi przed wskazanym momentem rozpoczęcia doby hotelowej. Przed przyjazdem do danego hotelu należy zapoznać się z obowiązującymi w nim godzinami rozpoczęcia oraz zakończenia doby hotelowej. W celu uzyskania większej ilości informacji na ten temat warto zaznajomić się z zawartością strony internetowej, załączonej w bibliografii niniejszej pracy [10].

2.7 Organizacja hotelu – recepcjoniści

Oczywistym jest, że organizacja obiektu hotelarskiego może być różna w zależności od ich wielkości oraz zakresu działalności, to znaczy przykładowo w przypadku małego pensjonatu, prowadzonego samodzielnie przez rodzinę, struktura organizacyjna jest klarowna i prosta, natomiast w przypadku dużego hotelu, wymagane jest zorganizowanie i utrzymywanie znacznie bardziej złożonej, rozbudowanej struktury organizacyjnej, a także zatrudnienie znacznie większej ilości pracowników. W celu dokładniejszego zapoznania się z procesem budowania struktury organizacyjnej hotelu, warto zapoznać się z zawartością strony internetowej, której odnośnik znajduje się w bibliografii [11]. W niniejszej pracy najbardziej interesującą częścią struktury organizacyjnej obiektu hotelarskiego jest recepcja. Podstawowym zadaniem recepcjonistów jest obsługa gościa hotelowego w trakcie jego zakwaterowania, a także pobytu. Jednakże do zadań recepcji hotelowej można zaliczyć również funkcje związane ze sprzedażą oraz marketingiem. Gdyż szczególnie w przypadku większych obiektów hotelarskich klient przede wszystkim utrzymuje kontakt z recepcją, z tego powodu niezwykle istotne jest wywarzenie pozytywnego wrażenia przez recepcjonistów.

2.8 Rezerwacja

Pojęcie rezerwacji należy rozumieć, jako pisemny lub ustny wniosek bądź umowę, z zawartą prośbą o zarezerwowanie konkretnej liczby miejsc noclegowych w wybranym przez klienta terminie. W czasach poprzedzających współczesną komputeryzację i automatyzację wszelkich usług, obsługą rezerwacji zajmował się recepcjonista. Jednakże aktualnie w niemal każdym obiekcie hotelarskim, może z wyłączeniem niektórych najmniejszych prowadzonych rodzinnie pensjonatów lub też gospodarstw

agroturystycznych, rezerwacja dokonywana jest przede wszystkim automatycznie przez potencjalnego klienta z wykorzystaniem odpowiedniej strony internetowej oraz aplikacji obsługującej działanie takiej strony. W celu dokonania rezerwacji, należy posiadać wiedzę na temat aktualnego obłożenia hotelu, a także ewentualnego wyłączenia z użytku danego pokoju z powodu chociażby remontu. Rezerwację zazwyczaj się dokonuje na konkretny typ pokoju, natomiast nie rezerwuje się konkretnego pokoju, jednakże taki sposób rezerwacji nie jest konieczny. Istnieje kilka sposobów na dokonanie rezerwacji, można do tego celu wykorzystać Internet lub też dokonać rezerwacji w bardziej tradycyjny sposób – telefonicznie lub bezpośrednio przebywając w danym hotelu. Także biuro usług pośredniczących lub inna instytucja może dokonać rezerwacji w imieniu klienta. Można wyróżnić następujące rodzaje rezerwacji:

- a) Rezerwacja niegwarantowana – jest to najbardziej standardowa rezerwacja, w dawniejszych czasach powszechnie stosowana, jednakże współcześnie coraz rzadziej jest ona stosowana. Klient opłacał taką rezerwację po pobycie, w momencie wymeldowania. Ta metoda rezerwacji jest coraz mniej popularna z powodu możliwości odwołania rezerwacji przez gościa hotelowego w ostatniej chwili bądź całkowitej rezygnacji z pobytu w hotelu bez uprzedniego poinformowania obsługi.
- b) Rezerwacja gwarantowana – to rezerwacja opłacona lub też zagwarantowana w inny sposób, dla przykładu kartą kredytową. Klient może zameldować się do końca pierwszej doby hotelowej bez utraty swojej rezerwacji. Istnieje także zwykle możliwość zrezygnowania z pobytu bez poniesienia kosztów przez potencjalnego gościa hotelowego, jednakże należy wówczas spełnić określone przez dany hotel warunki, zwykle wymagane jest anulowanie rezerwacji na kilka dni przed terminem rozpoczęcia pobytu w hotelu.

- c) Rezerwacja bezzwrotna – jest to najczęściej promocyjna oferta rezerwacji. Taka rezerwacja charakteryzuje się wymogiem w stosunku do klienta do dokonania całej opłaty za pobyt w momencie dokonania rezerwacji oraz brakiem możliwości rezygnacji wraz z uzyskaniem zwrotu środków finansowych.
- d) Rezerwacja wstępna – w przypadku tego rodzaju rezerwacji wymaga się od potencjalnego klienta potwierdzenia rezerwacji, zwykle poprzez zagwarantowanie pobytu pewną ilością środków finansowych, w jasno określonym terminie. W przypadku nie dokonania wymienionej powyżej czynności rezerwacja zostaje anulowana.

W celu zaznajomienia się z bardziej szczegółowym opisem rodzajów rezerwacji, istnieje możliwość zapoznania się z zawartością strony internetowej, której odnośnik znajduje się w bibliografii [12]. Możliwe jest także dokonanie podziału rezerwacji na rezerwację grupową i indywidualną. Rezerwacja indywidualna jest dokonywana przez osobę prywatną dla klienta indywidualnego, jednakże może być wykonana na więcej niż jednym pokoju. Z kolei rezerwacja grupowa jest dokonywana dla klienta grupowego przez konkretną instytucję, dla przykładu biuro turystyczne lub korporację.

2.9 Pokoje noclegowe

Pokoje hotelowe również dzieli się na rodzaje, poniżej zostały one wymienione:

- a) Jedyńka(ang. single room, skrót SGL) – jest to pokój dla jednego gościa z jednym łóżkiem.
- b) Podwójny dla jednej osoby (twin for sole use) – pokój z dwoma łózkami, jednakże przeznaczony jest dla jednej osoby.
- c) Podwójny (ang. twin room) – to pokój dwuosobowy, jednakże zawiera dwa pojedyncze łóżka.

- d) Dwójka (ang. double room, skrót DBL) – to pokój przeznaczony dla dwóch osób, posiadający jedno dwuosobowe łóżko, zwane „małżeńskim”.
- e) Trójka (ang. triple) – pokój trzyosobowy, zazwyczaj zawierający trzy pojedyncze łóżka.
- f) Czwórka (ang. quad) – pokój czteroosobowy.
- g) Pokoje typu „dormitory” – są to pokoje zawierające wiele łóżek, które są wynajmowane różnym gościom.

W celu bardziej dogłębnego zapoznania się z poruszaną tutaj dziedziną, warto zaznajomić się ze stroną internetową opisującą dokładniej pokoje hotelowe, odnośnik znajduje się w bibliografii [13].

2.10 Systemy do obsługi hotelu

W tym podrozdziale omówione zostały systemy przeznaczone do obsługi hotelu, aktualnie znajdujące się na rynku. Przed omówieniem poszczególnych systemów warto zaznajomić się z zagadnieniem systemów CRS (ang. Computerised Reservation Systems – komputerowe systemy rezerwacji). Określenia tego używa się w stosunku do klasy programów informatycznych, odpowiadających za przechowywanie oraz odzyskiwanie informacji dotyczących usług oferowanych przez daną firmę, a także do dokonywania sprzedaży tychże usług za pośrednictwem Internetu. Tego typu systemy zazwyczaj są własnością firm świadczących usługi hotelarskie. Umożliwiają one sprzedaż między innymi pokoi hotelowych. Z kolei systemy dysponujące zasięgiem globalnym nazywa się systemami GDS (Global Distribution Systems – globalny system dystrybucji). Większość, szczególnie większych ośrodków hotelarskich korzysta z jednego lub też z większej ilości systemów

CRS. W Internecie znajdują się opisy oraz oferty ogromnej ilości systemów do obsługi hotelu. Trzy wybrane zostaną teraz pokrótce opisane:

- a) KWhotel jest systemem zarządzania hotelem. Istnieje możliwość nabycia tego produktu w wersji darmowej, jednostanowiskowej oraz wielostanowiskowej. Oprogramowanie to zostało zintegrowane z portalem booking.com i channel managerami. Charakterystyczną funkcjonalnością tego systemu jest współpraca z większością zamków hotelowych dostępnych na rynku, co jest powodem ogromnej popularności tego programu. KWhotel znajduje się w czołówce wśród programów wybieranych w celu nadzoru nad apartamentami.
- b) izzyHotel to kolejny system przeznaczony do zarządzania ośrodkiem hotelowym, a także ośrodkiem konferencyjnym. Program ten jest następcą znanego oprogramowania HotelOnLine. Oprogramowanie to zostało oparte na bazie dwunastu lat doświadczeń w dziedzinie tworzenia systemów do obsługi hoteli. Wykorzystano dotychczasowe doświadczenia oraz oczekiwania klientów, a także możliwości oferowane przez nowe technologie. W systemie tym, pracującym pod kontrolą systemu operacyjnego Windows, zawarto między innymi mechanizm automatycznych kopii bezpieczeństwa oraz wbudowany moduł tworzenia raportów i zestawień. Istnieje także możliwość współpracy z portalami internetowymi, takimi jak profitroom.pl.
- c) X2System dla hoteli to pakiet oprogramowania, w składzie którego można wyróżnić oprogramowanie dedykowane X2Hotel. Z powodu modułowej budowy istnieje możliwość dostosowania systemu do specyficznych potrzeb konkretnego hotelu oraz integracji z pozostałymi pakietami oprogramowania z rodziny X2System, między innymi z programem obsługującym gastronomię.

3. Wymagania i narzędzia

W rozdziale tym opisane zostały wymagania postawione przed aplikacją, zarówno funkcjonalne, jak i нефункционаłne, a także przypadki użycia przedstawione za pomocą diagramu UML (ang. Unified Modeling Language - zunifikowany język modelowania) oraz krótki opis architektury aplikacji – klient-serwer. Opisane zostały narzędzia użyte w programie, a także metodyka pracy nad projektowaniem oraz implementacją aplikacji.

3.1 Wymagania funkcjonalne

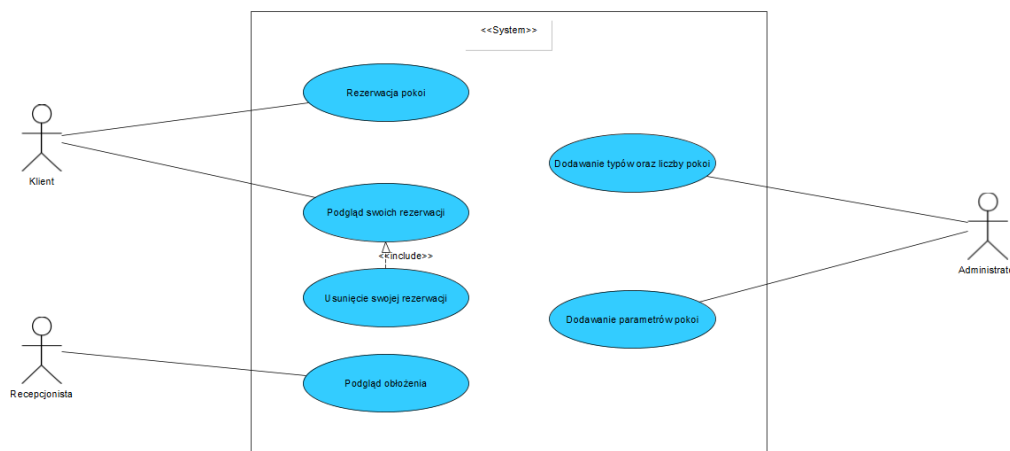
- a) Podstawową funkcjonalność systemu stanowi możliwość dokonania rezerwacji pokoju hotelowego poprzez stronę internetową, z uwzględnieniem wyboru terminu pobytu, rodzaju pokoju oraz wszystkich opcji oferowanych przez hotel.
- b) Z kolei recepcjonista pomijając możliwość rezerwacji pokoju na podane przez klienta nazwisko ma możliwość podglądu obłożenia oraz usuwania rezerwacji dokonanych przez klientów.
- c) Administrator ma możliwość dokonania edycji parametrów pokoi w celu dostosowania ich do konkretnego hotelu, a także dodania lub usunięcia pokoi oraz typów pokoi.

3.2 Wymagania нефunkcjonalne

- a) Hasła w bazie danych są przechowywane w postaci zaszyfrowanej.
- b) Dane użytkowników, usuwających konto, są usunięte.
- c) Dane użytkowników nie są przekazywane podmiotom trzecim.
- d) Aplikacja została przystosowana do pracy pod kontrolą wszystkich wiodących przeglądarek internetowych.
- e) Istnieje możliwość dostosowania interfejsu aplikacji przeglądarkowej do różnych rodzajów ekranów.
- f) Dalszy rozwój aplikacji jest możliwy, nowe funkcjonalności mogą zostać łatwo dodane.

3.3 Przypadek użycia – diagram UML

Na diagramie przypadków użycia w języku UML (Rys. 3.1) zobrazowany został przypadek użycia systemu po dokonaniu logowania użytkownika do programu:



Rys. 3.1: Diagram przypadków użycia systemu.

3.4 Opis narzędzi użytych w programie

Aplikacja składa się z dwóch całkowicie niezależnych od siebie części powiązanych tylko poprzez API (ang. application programming interface - interfejs programowania aplikacji) udostępniane przez serwer aplikacji. Ten komponent systemu został napisany w języku Java z wykorzystaniem szkieletu tworzenia aplikacji Spring, służy on również do połączenia się z bazą danych PostgreSQL wykorzystywaną do przechowywania informacji. Druga część systemu – klient aplikacji napisany został w języku Typescript oraz HTML (ang. HyperText Markup Language - hipertekstowy język znaczników) z wykorzystaniem platformy internetowej Angular.

Przechodząc do opisu technologii Spring, jest to platforma dedykowana do tworzenia aplikacji w języku Java, składająca się z wielu osobnych projektów. Kontener wstrzykiwania zależności stanowi kluczowy element

tego systemu, jednakże na przestrzeni lat Spring uzyskał wsparcie dla wielu technologii, dzięki czemu stanowi dziś jeden z głównych komponentów całego ekosystemu Javy. Można wręcz powiedzieć, że zaistniała swego rodzaju rywalizacja pomiędzy technologią Spring, a Java Enterprise Edition, z tego powodu we współczesnych wersjach tych technologii widać bardzo podobne podejście, przenikanie się obu rozwiązań, a także wzajemne inspiracje. Spring składa się obecnie z wielu osobnych modułów, z których parę wykorzystanych w projekcie zostanie teraz krótko opisanych: Spring framework jest kluczowym elementem całego szablonu Spring, udostępniającym kontener wstrzykiwania zależności. Spring Boot stanowi tak naprawdę alternatywę dla Javy Enterprise Edition, i umożliwia zautomatyzowanie oraz ukrycie przed użytkownikiem skomplikowanego procesu konfiguracji tworzonej aplikacji. Na podstawie skanowania pakietów oraz informacji zawartych w pliku `application.properties`, Spring Boot zajmuje się tworzeniem niezbędnych obiektów (np. `EntityManagerFactory` dla JPA). Spring Data REST (ang. Representational state transfer - styl architektury oprogramowania wykorzystywany przy tworzeniu aplikacji internetowych) znacznie ułatwia budowanie usług internetowych opartych na architekturze REST API. Spring Data JPA (ang. Java Persistence API - oficjalny standard mapowania obiektowo-relacyjnego dla języka programowania Java) jest projektem upraszczającym dostęp do baz danych. Główną zaletą tej technologii jest zminimalizowanie ilości powtarzalnego kodu. Dzięki modułowi JPA możliwe jest korzystanie z interfejsu `CrudRepository` do obsługi podstawowych metod CRUD (ang. create - stwórz, read - odczytaj, update - zaktualizuj, delete - usuń). Spring MVC (ang. Model View Controller – model, widok, kontroler). jest projektem będącym najpopularniejszym szkieletem zgodnym ze wzorcem architektonicznym MVC, wykorzystywanym do tworzenia aplikacji webowych w Javie. Stanowi alternatywę dla JSF (ang. Java Server Faces – szkielet upraszczający tworzenie interfejsu użytkownika aplikacji) z Javy EE (Enterprise Edition). Spring MVC bazuje na technologii serwletów, czyli kluczowej specyfikacji Javy EE i do działania wymaga kontenera serwletów typu Tomcat. Spring

Security to projekt pozwalający na zabezpieczenie aplikacji i stworzenie systemu uwierzytelnienia oraz autoryzacji.

W tym miejscu zostanie opisana technologia wykorzystana przy implementacji klienta aplikacji. Ta część systemu została stworzona z użyciem platformy internetowej Angular, stanowiąca otwarte środowisko programistyczne, a także platformę do tworzenia SPA (ang. Single Page Application – jednostronicowa aplikacja internetowa - posiadającą tylko jeden plik html), napisaną w języku TypeScript i wspierany oraz rozwijany przez Google. Początkowo Angular stanowił bibliotekę języka JavaScript, znaną pod nazwą AngularJS, jednak po zastosowaniu znacznych modyfikacji od wersji drugiej, technologia ta wyewoluowała jako całkowicie odrębna platforma programistyczna. Po tym czasie wykorzystuje się język TypeScript, będący tak naprawdę semantycznie nadzbiorem języka JavaScript. Składnia obu tych języków jest bardzo do siebie zbliżona, technologia TypeScript oferuje dodatkowe możliwości. W celu uruchomienia aplikacji wykorzystano CLI (ang. command line interpreter/interface – wiersz poleceń) Angulara oraz Node.js. Angular CLI to narzędzie wiersza poleceń, stworzone w celu ułatwienia konfiguracji nowych projektów tworzonych w najnowszych wersjach Angulara. Node.js, to wieloplatformowe środowisko uruchomieniowe o otwartym kodzie do tworzenia aplikacji typu server-side napisanych w języku JavaScript. W aplikacji wykorzystano również bibliotekę CSS (ang. Cascading Style Sheets – język służący do opisu formy prezentacji (wyświetlania) stron WWW) o nazwie Bootstrap – jest to gotowy szkielet do budowania wizualnej części stron internetowych. Z kolei Angular Material stanowi zestaw komponentów implementujących specyfikację Google Material Design, wykorzystany w celu ułatwienia procesu tworzenia widoku aplikacji. Wykorzystano także technologię HTML oraz CSS do utworzenia widoków komponentów.

3.5 Opis pracy nad projektowaniem i implementacją systemu

Proces projektowania oraz implementacji systemu do obsługi hotelu przebiegał w sposób następujący: rozpoczęto od zapoznania się ze sposobem budowy, funkcjonowania dostępnych w Internecie rozwiązań, a także ich architekturą, w szczególności z architekturą Klient-Serwer. Następnie zostało przeanalizowane działanie technologii wybranych podczas projektowania systemu: Spring po stronie serwera oraz Angular po stronie klienta aplikacji. W celu bardziej dogłębnego zapoznania się ze sposobem funkcjonowania tych technologii zapoznano się z zawartością paru książek [1-3]. Kolejnym etapem było przygotowanie wstępnej implementacji systemu z wykorzystaniem wybranych technologii oraz dopracowywanie finalnego efektu pracy.

4. Specyfikacja zewnętrzna

W rozdziale tym opisany został sposób działania systemu oraz funkcji dostępnych w aplikacji. Przedstawione zostały wymagania sprzętowe systemu oraz sposób jego instalacji, a także uruchomienia. Wyszczególnione zostały kategorie użytkowników dostępnych w systemie oraz sposób obsługi programu i administracja systemem. Szczególną uwagę poświęcono kwestiom bezpieczeństwa. Na koniec zobrazowano przykładowe działanie programu, wykorzystując zrzuty ekranu aplikacji wraz z różnymi scenariuszami korzystania z aplikacji.

4.1 Wymagania sprzętowe oraz programowe systemu

Aby uruchomić program należy posiadać komputer wraz z zainstalowaną przeglądarką internetową, a także uzyskać dostęp do Internetu. W celu instalacji aplikacji wymagane jest posiadanie zainstalowanego środowiska uruchomieniowego Javy (ang. Java Runtime Environment – JRE) do uruchomienia serwera systemu. Konieczne jest także posiadanie zainstalowanego oprogramowania node.js będącego biblioteką uruchomieniową języka JavaScript, niezbędną do uruchomienia klienta danej aplikacji. Do działania programu potrzebne jest ponadto oprogramowanie systemu zarządzania bazą danych PostgreSQL. Wszystkie te narzędzia są dostępne do pobrania na ich własnych stronach internetowych.

4.2 Sposób instalacji i aktywacji

Przed rozpoczęciem użytkowania aplikacji należy zainstalować oprogramowanie wspomniane w poprzednim podrozdziale: JRE, node.js oraz PostgreSQL. Proces instalacji systemu PostgreSQL jest zdecydowanie najbardziej skomplikowany, należy odpowiednio skonfigurować połączenie na porcie 5433, pozostałe właściwości powinny pozostać domyślne. Proces instalacji aplikacji rozpoczyna się od rozpakowania do dowolnej lokalizacji dysku twardego komputera pliku zawierającego spakowane komponenty niezbędne do zainstalowania systemu. Po rozpakowaniu należy przejść do wnętrza folderu „aplikacja do obsługi hotelu”, a następnie dwukrotnie uruchomić wiersz poleceń systemu w danej lokalizacji. W jednym oknie polecenia należy wpisać komendę: „java -jar hotel.jar”. Po uruchomieniu danej komendy powinien zacząć uruchamiać się serwer (na początku powinno być widoczne w oknie polecenia logo Spring’a). Z kolei w drugim należy uruchomić klienta aplikacji. Do tego celu niezbędne jest posiadanie zainstalowanego oprogramowania node.js, możliwe jest zorientowanie się czy na danym komputerze to oprogramowanie jest zainstalowane poprzez podanie komendy: „npm --version” (jeśli jest zainstalowane, to pojawi się numer wersji zainstalowanej na danym komputerze). Uruchomienie narzędzia Angular CLI odbywa się po podaniu komendy „ng serve --open”. Jednakże bez uprzedniego zainstalowania Angular CLI aplikacja nie zostanie poprawnie uruchomiona, można tego dokonać za pomocą komendy „npm install”, jeżeli po wykonaniu wszystkich tych komend wciąż nie będzie możliwe uruchomienie klienta aplikacji, można wykonać również komendę: „npm update”. Po wykonaniu tych czynności aplikacja powinna być gotowa do działania.

4.3 Kategorie użytkowników dostępnych w systemie

W systemie znajdują się trzy kategorie użytkowników. Pierwszą kategorią jest administrator systemu, on ma największe uprawnienia, może robić wszystko to, co pozostałe kategorie użytkowników, do tego jako jedyny ma uprawnienia do zarządzania parametrami pokoi, do zmieniania dla przykładu liczby pokoi danego typu albo dodawania nowych pokoi czy wyłączania z użytku takich, które przestają być dostępne dla klientów. Kolejnym rodzajem użytkowników są recepcjoniści hotelu. Mają oni podgląd obłożenia hotelu, mogą zaznaczyć w systemie, że dana rezerwacja została opłacona albo usunąć daną rezerwację. Pozostali użytkownicy są klientami, system udostępnia takim użytkownikom możliwość rezerwacji pokoju na dowolny termin. Program automatycznie sprawdza, czy w danym terminie są wolne pokoje danego typu i tylko w takim wypadku umożliwia dokonanie na dany typ pokoju rezerwacji. Klient ma również możliwość podglądu swoich rezerwacji, a także anulowania swoich rezerwacji.

4.4 Sposób obsługi programu

W celu rozpoczęcia korzystania z aplikacji najpierw należy się zalogować, poprzez podanie swojego loginu oraz hasła w panelu logowania. Zalogowanie się jest konieczne, aby uzyskać możliwość dokonania rezerwacji, a także dostęp do pozostałych funkcjonalności udostępnianych przez ten program. Zatem w celu dokonania rezerwacji wymagane jest posiadanie konta w systemie, w przypadku gdy takiego konta nie ma, klient ma możliwość zarejestrowania się na stronie internetowej aplikacji. Konto dla recepcjonisty może zostać dodane przez administratora systemu. Rezerwacji pokoju, podglądu/edycji swoich rezerwacji czy też podglądu obłożenia

dokonywane jest poprzez kliknięcie w odpowiedni przycisk przenoszący do odpowiedniego modułu aplikacji odpowiedzialnego za daną czynność. Przykład działania aplikacji wraz ze zrzutami ekranu ilustrującymi działanie programu został przedstawiony w jednym z kolejnych podrozdziałów tego rozdziału.

4.5 Administracja systemem

Administrator zajmuje się administracją systemem. Po zalogowaniu się na konto administratora systemu, możliwe staje się zarządzanie parametrami poszczególnych rodzajów pokoi, dodawanie/edycja właściwości pokoi, a także zarządzanie użytkownikami – dodawanie nowych recepcjonistów/klientów i usuwanie ich z systemu. W przypadku wystąpienia jakichkolwiek problemów z działaniem aplikacji, w celu zweryfikowania poprawności zapisu danych do bazy danych systemu istnieje możliwość uruchomienia programu pgAdmin przez administratora systemu sprawdzenie czy dane użytkowników, rezerwacji, pokoi oraz ich parametrów zostały właściwie zapisane w bazie danych.

4.6 Kwestie bezpieczeństwa

Dostęp do poszczególnych części aplikacji jest możliwy jedynie dla zalogowanych użytkowników. Z kolei sam proces uzyskiwania zasobów z serwera programu jest chroniony za pomocą autoryzacji z wykorzystaniem tokenu JWT (ang. JSON (ang. JavaScript Object Notation – lekki format

wymiany danych komputerowych) Web Token). Dzięki temu żadna inna aplikacja poza klientem dedykowanym do obsługi serwera aplikacji nie ma dostępu do żadnych zasobów, w przypadku próby pozyskania takowych zasobów z serwera, dla przykładu w przeglądarce internetowej wyskakuje błąd HTTP (ang. Hypertext Transfer Protocol - protokół przesyłania dokumentów hipertekstowych) numer 401, który oznacza brak autoryzacji (ang. unauthorized). Także niektóre moduły programu są dostępne tylko dla administratora systemu lub też dla recepcjonistów, w przypadku próby do nich dostępu na przykład przez klienta, tzw. strażnik aplikacji nie pozwoli na otwarcie się żądanej strony.

4.7 Scenariusze korzystania z systemu - przykład działania aplikacji

Aplikacja może być obsługiwana zarówno przez administratora, recepcjonistę, jak i przez zwykłego klienta. Najpierw jednak wymagane jest zalogowanie się do aplikacji, po jej uruchomieniu widoczny jest panel logowania (*Rys. 4.1*).

X zamknij panel rezerwacji

Podaj termin rezerwacji

Zarezerwuj pokój

Witaj na stronie hotelowej! Nie jesteś zalogowany - zaloguj się!

nazwa użytkownika

Hasło

Zaloguj się

Pamiętaj mnie

Nie masz konta? Zarejestruj się

Zapomniałem hasła

Wróć do strony głównej

Witaj na stronie przeznaczonej do rezerwacji pokoi w hotelu!

Rys. 4.1: Ekran widoczny po uruchomieniu aplikacji.

W przypadku gdy zostanie wprowadzona w panelu bocznym aplikacji data początku oraz końca rezerwacji, po czym zostanie naciśnięty przycisk „Zarezerwuj pokój” bez uprzedniego zalogowania się, zostanie wyświetlony komunikat o konieczności zalogowania się w celu rezerwacji pokoju oraz nastąpi automatyczne otwarcie widoku rejestracji nowego użytkownika (Rys. 4.2).

The screenshot shows a web application interface for a hotel reservation system. At the top, a dark header bar contains the text "Witaj na stronie hotelowej! Nie jesteś zalogowany - zaloguj się!". Below this, there are two input fields: "nazwa użytkownika" (username) and "hasło" (password), followed by a "Zaloguj się" (Login) button and a "Pamiętaj mnie" (Remember me) checkbox. A red error message states: "Nie masz konta? Zarejestruj się" (Don't have an account? Register now). Below this, a pink banner reads: "Musisz być zalogowany w systemie, aby dokonać rezerwacji, jeśli nie masz konta, zarejestruj się teraz!" (You must be logged in to the system to make a reservation, if you don't have an account, register now!). A link "Wróć do strony głównej" (Return to the main page) is visible. The main content area is titled "Rejestracja nowego klienta" (New client registration) and contains five input fields: "imię" (first name), "nazwisko" (last name), "login", "hasło", and "email". At the bottom right of this section are two buttons: "Dodaj użytkownika" (Add user) and "Anuluj" (Cancel). On the left side of the page, there is a sidebar with a "Zarezerwuj pokój" (Reserve room) button and a notification "X zamknij panel rezerwacji" (Close reservation panel).

Rys. 4.2: Ekran widoczny po próbie rezerwacji przez niezarejestrowanego użytkownika.

W przypadku gdy zostanie wykonana operacja zalogowania się, ekran aplikacji, a także dostępne scenariusze korzystania z systemu będą się różnić w zależności od kategorii zalogowanego użytkownika. W przypadku administratora po zalogowaniu się pojawi się ekran wyboru jednej z opcji administracyjnych (Rys. 4.3).

The screenshot shows the dashboard of a hotel reservation system for a logged-in administrator. The header bar is dark and contains the text "Zalogowano jako Główny Administrator" (Logged in as Main Administrator) and a "Wyloguj się" (Logout) button. Below the header, there is a link "Wróć do strony głównej" (Return to the main page). The main content area is titled "wybierz, co chcesz zrobić:" (choose what you want to do:). It contains five buttons: "Przejdź do panelu zarządzania parametrami pokoi" (Go to the room parameter management panel), "Przejdź do panelu zarządzania typami i ilością pokoi" (Go to the room types and quantity management panel), "Przejdź do panelu zarządzania użytkownikami" (Go to the user management panel), "Przejdź do podglądu obłożenia" (Go to the occupancy view), and "Przejdź do podglądu Twoich rezerwacji" (Go to the view of your reservations).

Rys. 4.3: Ekran widoczny po zalogowaniu się jako administrator.

Po wybraniu przez administratora opcji przejścia do panelu zarządzania parametrami pokoi nastąpi przejście do panelu, w którym istnieje możliwość podglądu a także edycji parametrów dostępnych pokoi w systemie (Rys. 4.4).

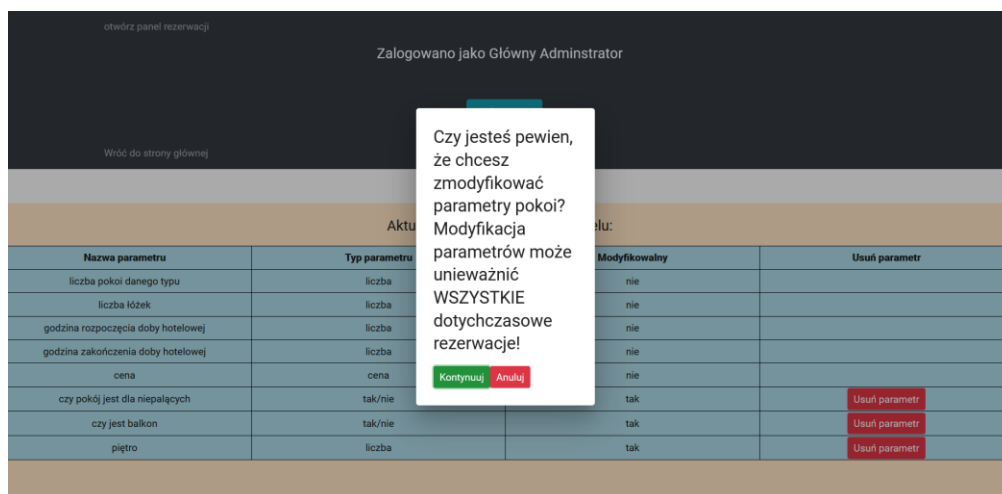


The screenshot shows a web interface for a hotel management system. At the top, there is a dark header bar with the text 'Zalogowano jako Główny Administrator' and a 'Wyloguj się' button. Below the header, there is a light orange bar with the text 'Aktualne parametry pokoi w Hotelu:'. Underneath this bar is a table with three columns: 'Nazwa parametru', 'Typ parametru', and 'Modyfikowalny'. The table contains eight rows of data. At the bottom of the table, there is a light orange bar with an 'Edytuj parametry' button.

Nazwa parametru	Typ parametru	Modyfikowalny
liczba pokoi danego typu	liczba	nie
liczba łóżek	liczba	nie
godzina rozpoczęcia doby hotelowej	liczba	nie
godzina zakończenia doby hotelowej	liczba	nie
cena	cena	nie
czy pokój jest dla niepalących	tak/nie	tak
czy jest balkon	tak/nie	tak
piętro	liczba	tak

Rys. 4.4: Ekran widoczny po przejściu do panelu edycji parametrów.

W momencie kliknięcia przycisku „edytuj parametry” pojawi się okno ostrzegające użytkownika o możliwości utraty wszystkich dotychczasowych rezerwacji w systemie w przypadku edycji dostępnych parametrów (Rys. 4.5).



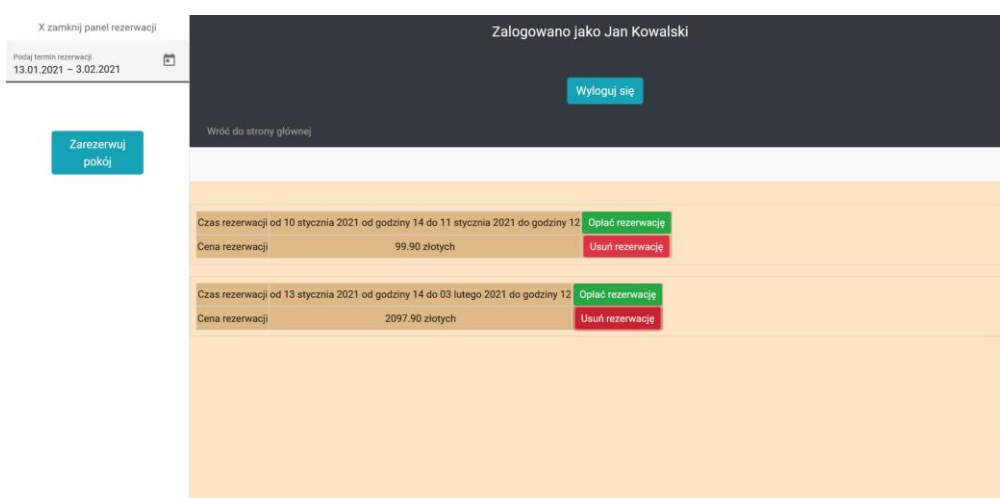
Rys. 4.5: Ekran widoczny po wciśnięciu przycisku „edytuj parametry”.

W przypadku wybrania opcji „Anuluj” nastąpi powrót do głównego panelu administracyjnego, natomiast w przypadku wybrania opcji kontynuuj pojawi się możliwość dodawania oraz usuwania parametrów (Rys. 4.6).



Rys. 4.6: Ekran widoczny po wciśnięciu przycisku „kontynuuj”.

W aplikacji istnieje więcej możliwości dostępnych dla administratora systemu. W przypadku recepcjonisty, po zalogowaniu się możliwe jest przejście do podglądu obłożenia, gdzie istnieje opcja usunięcia rezerwacji dla poszczególnych klientów. Z kolei analizując scenariusz klienta, po wybraniu opcji poglądu rezerwacji ukazuje się prosty widok dokonanych rezerwacji przez klienta wraz z opcją rezygnacji z nich lub też odnośnik to ich opłacenia (Rys. 4.7).



Rys. 4.7: Ekran widoczny po wciśnięciu przycisku „przejdź do podglądu twoich rezerwacji”.

W przypadku wybrania opcji rezerwacji pokoju po wybraniu terminu rezerwacji, następuje przejście do panelu wyboru typu pokoju, gdzie widoczne są jedynie dostępne pokoje w danym terminie (Rys. 4.8).

Dobierz dla siebie odpowiedni pokój:

Wybierz ilościowy pokój chcesz zamówić *

2

czy pokój jest dla niepalących *

tak

czy jest balkon *

tak

piętro *

3

Przejdź dalej Wróć

Rys. 4.8: Ekran widoczny po wciśnięciu przycisku „zarezerwuj pokój”.

Po przejściu dalej widoczne jest podsumowanie rezerwacji ze wszystkimi danymi rezerwacji(Rys. 4.9).

Zalogowano jako Jan Kowalski

Wyloguj się

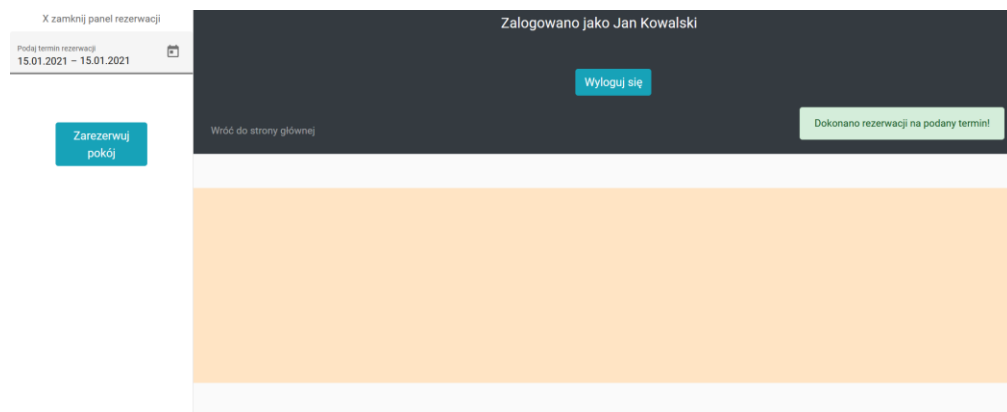
Wróć do strony głównej

Nazwa typu pokoju	Typ wybrany w rezerwacji
Czas rezerwacji	od 15 stycznia 2021 od rozpoczęcia doby hotelowej (godzina 14) do 16 stycznia 2021 do zakończenia doby hotelowej (godzina 12)
Liczba łóżek w pokoju	1
godzina rozpoczęcia doby hotelowej	14
godzina zakończenia doby hotelowej	12
piętro	3
czy pokój jest dla niepalących	tak
czy jest balkon	tak
Cena rezerwacji	99.90 złotych

Zarezerwuj pokój Wróć

Rys.4.9: Ekran widoczny po wciśnięciu przycisku „przejdź dalej”.

Po wybraniu opcji „zarezerwuj pokój” powinna nastąpić rezerwacja pokoju, w takim przypadku pojawi się odpowiedni komunikat o dokonaniu rezerwacji, a także nastąpi powrót do strony głównej (Rys. 4.10).



Rys. 4.10: Ekran widoczny po wciśnięciu przycisku „zarezerwuj pokój”.

5. Specyfikacja wewnętrzna

W rozdziale tym opisana została idea działania aplikacji, a także architektura systemu. Również opisane zostały użyte w programie struktury danych oraz organizacja bazy danych. Opisano także użyte w programie komponenty, moduły, biblioteki, ważniejsze klasy i algorytmy. Wybrane fragmenty kodu zostały szczegółowo opisane.

5.1 Idea działania aplikacji

Zgodnie z założeniem, dane wprowadzane przez użytkownika powinny zostać przekazane od klienta aplikacji do serwera, którego zadaniem jest zapisanie takowych danych w wykorzystywanej bazie danych. Dane konieczne do obsługi programu są pobierane przez serwer systemu, a także udostępniane klientowi aplikacji. Komunikacja pomiędzy obiema częściami systemu powinna być zaszyfrowana.

5.2 Architektura systemu

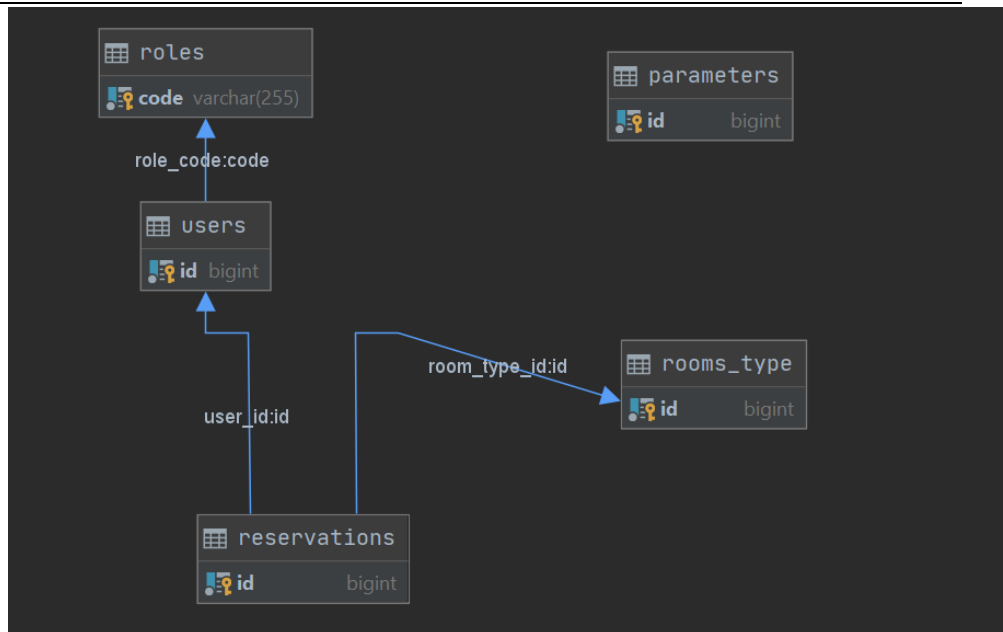
Podczas projektowania systemu oparto się na wzorcu architektonicznym MVC. Wyróżnia się trzy warstwy takiego wzorca:

- a) Model – programowe odwzorowanie pewnego fragmentu danego problemu.
- b) View (widok) – wizualizacja graficznego interfejsu użytkownika prezentująca dane aplikacji.
- c) Controller (kontroler) – warstwa pośrednicząca, odpowiadająca za komunikację pomiędzy modelem oraz widokiem, a także umożliwiającą interakcję użytkownika.

Zachowano zgodność ze stylem architektury oprogramowania REST API zaproponowanym przez Roya T. Fieldinga w 2000 roku. W celu uzyskania dokładniejszych informacji na temat stylu REST API czym się charakteryzuje, należy zapoznać się z informacjami dostępnymi w pracy dostępnej pod adresem internetowym podanym w bibliografii [14]. Warto podkreślić, że istnieje model dojrzałości Richardsona służący do sklasyfikowania oprogramowania zgodnego z wymienionym wyżej stylem, dzielący REST API na cztery różne poziomy [15]. Opisywana w tej pracy aplikacja zachowuje zgodność z poziomem drugim rzeczony klasyfikacji.

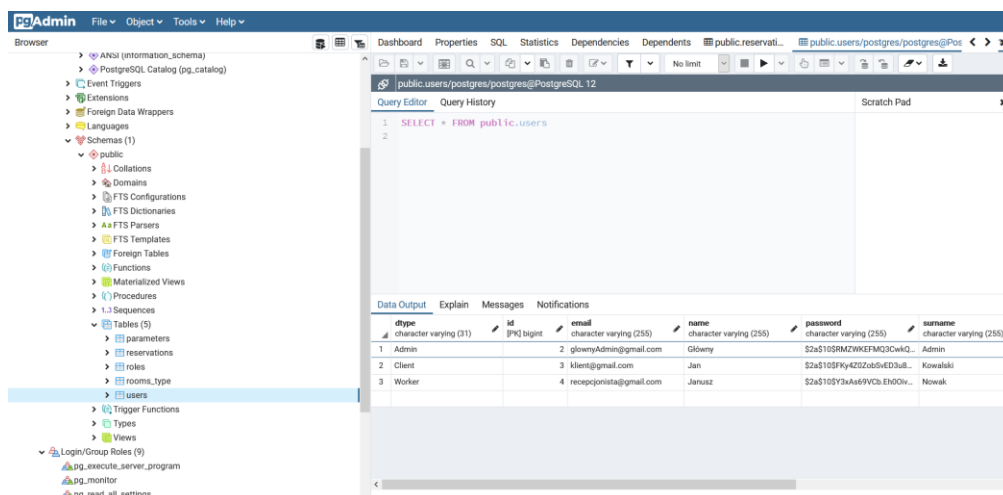
5.3 Opis użytych struktur danych, a także organizacji bazy danych

Schemat bazy danych programu składający się z pięciu tabel przedstawiono na rysunku (Rys. 5.1).



Rys. 5.1: Zrzut ekranu przedstawiający schemat bazy danych systemu.

Dane potrzebne do działania programu są przechowywane w systemie zarządzania bazą danych PostgreSQL. Istnieje możliwość podglądu zapisanych danych w bazie oraz ich ewentualnej edycji lub edycji struktury bazy danych za pomocą systemu pgAdmin 4 (Rys.5.2).

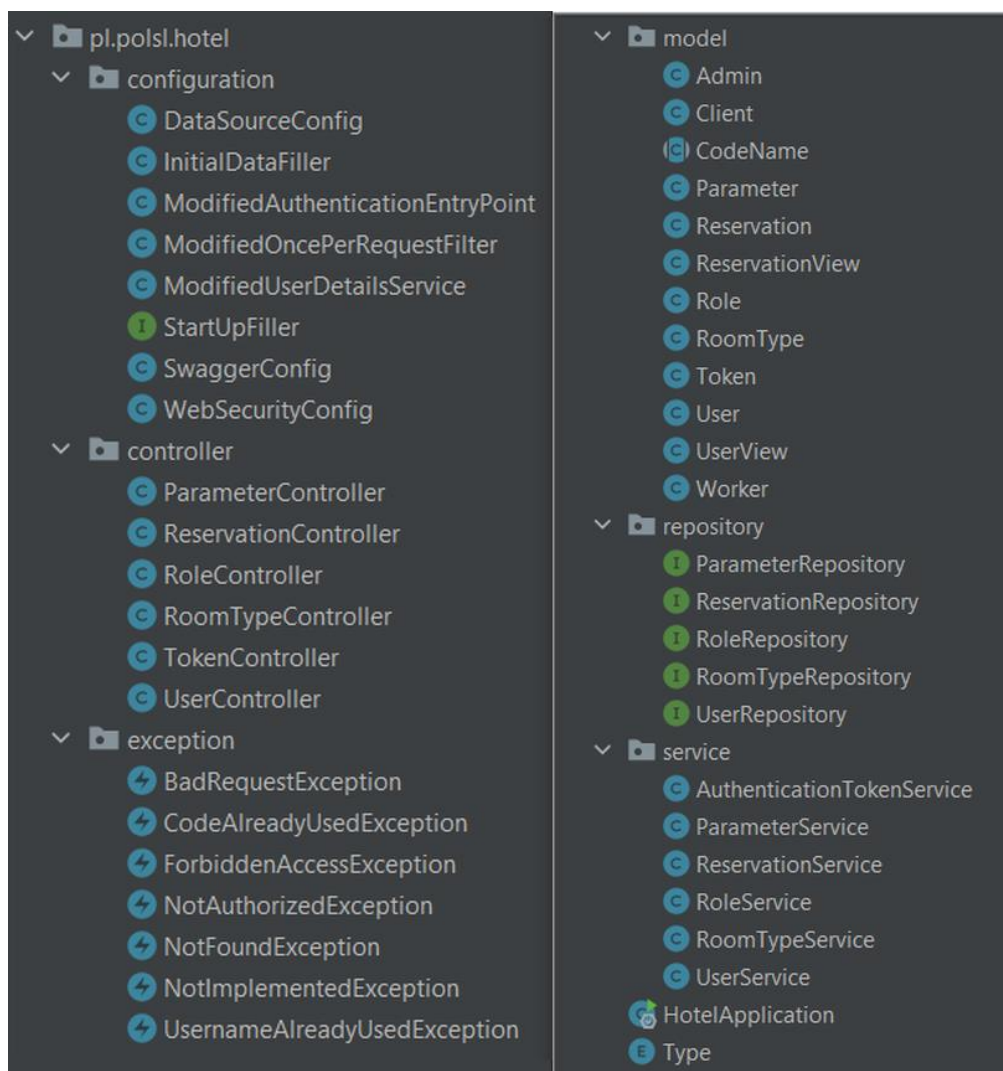


Rys. 5.2: Przykładowy zrzut ekranu przedstawiający widok panelu administratora pgAdmin4.

Dane zapisywane są do bazy danych, a także odczytywane z tejże bazy za pomocą interfejsu JPA, konkretnie implementacji Hibernate. Jest to biblioteka zapewniająca dostęp do danych zapisanych w relacyjnej bazie danych, oraz mapowanie obiektowo-relacyjne (ORM – ang. Object-Relational Mapping). W celu bardziej dogłębnego zapoznania się z technologią Hibernate istnieje możliwość pozyskania informacji ze źródła internetowego [16]. Informacje pomiędzy serwerem oraz klientem aplikacji są wymieniane w postaci formatu JSON (JavaScript Object Notation).

5.4 Opis wykorzystanych komponentów, modułów, bibliotek, ważniejszych klas użytych w programie

Struktura katalogów w serwerze systemu oraz nazwy wykorzystanych komponentów zostały przedstawione na zrzucie ekranu (Rys.5.2).



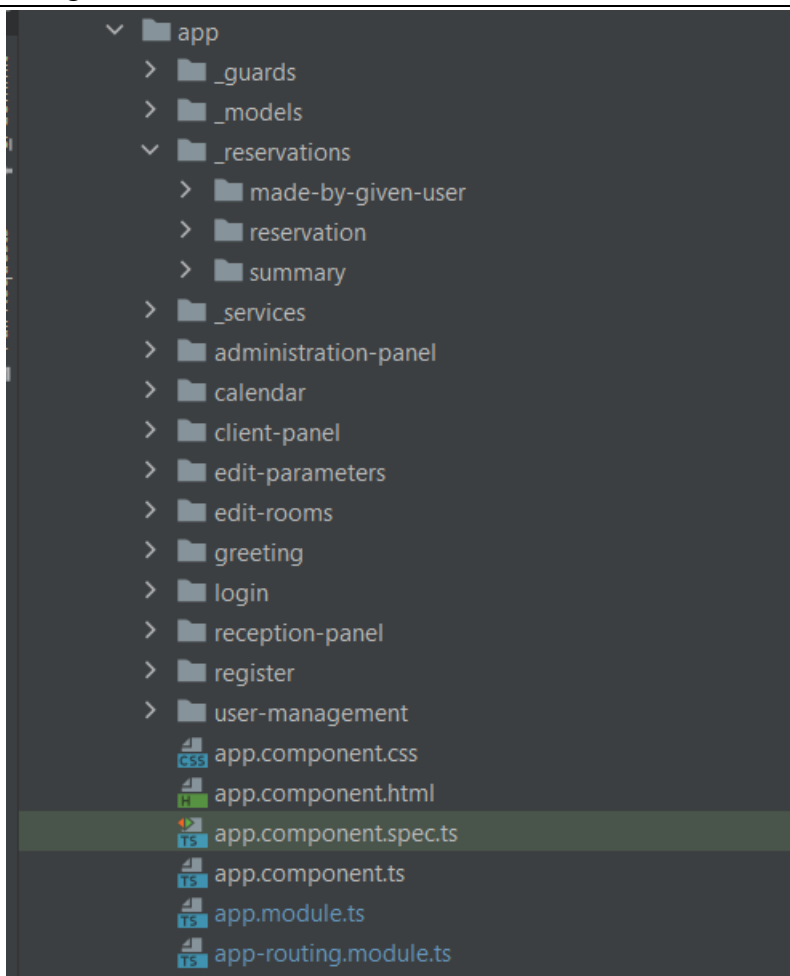
Rys. 5.2: Zrzut ekranu przedstawiający strukturę katalogów w serwerze aplikacji.

Widoczny jest podział serwera aplikacji na warstwy odpowiadające za poszczególne zadania. Każda warstwa programu znajduje się w osobnym pakiecie:

- a) Configuration (konfiguracja) – pakiet zawierający klasy konfiguracyjne serwera systemu.

- b) Controller (kontroler) – można określić ten pakiet jako warstwę aplikacji, gdzie wywołuje się odpowiednie serwisy oraz przetwarza zapytania HTTP.
- c) Exception (wyjątek) – pakiet zawierający klasy implementujące odpowiednie wyjątki umożliwiające wyświetlenie odpowiedniego statusu błędu zgodnego z protokołem HTTP.
- d) Model – pakiet zawierający klasy modelujące jakieś obiekty.
- e) Repository (repozytorium) – pakiet zawierający interfejsy rozszerzające interfejsy JPA, umożliwiające zapis do bazy danych, a także odczyt jej zawartości – warstwa dostępu do danych.
- f) Service (Serwis) – pakiet serwisów wywoływanych w odpowiednim kontrolerze, w tym miejscu następuje wywołanie odpowiednich metod z repozytorium obsługującego bazę danych oraz odpowiednie przetwarzanie danych – warstwa logiki biznesowej.

Z kolei struktura katalogów, a także nazwy komponentów wykorzystanych w kliencie aplikacji zostały przedstawione na kolejnym zrzucie ekranu (Rys.5.3).



Rys. 5.3: Zrzut ekranu przedstawiający strukturę katalogów w kliencie aplikacji.

5.5 Szczegółowy opis implementacji wybranych fragmentów programu

W listingu 5.1 przedstawiony został fragment kodu serwera aplikacji, konkretnie klasy `RoomTypeService` w pakiecie `service`.

Listing 5.1. Funkcja zwracająca wolne pokoje w danym okresie czasu.

```
1  public List<RoomType> getRoomsAvailable
2  (String start, String end)
3  {
4      //wywołanie funkcji konwertującej daty do odpowiedniego
5      //typu
6      convertDates(start, end);
7
8      //pętla po wszystkich typach pokoi w bazie danych
9      for (RoomType roomType : roomTypeRepository.findAll())
10     {
11         isAvailable = true;
12         numberOfRoomsAvailable = roomType.getNumber1();
13         //pętla po wszystkich typach rezerwacji
14         for (Reservation reservation : reservationRepository
15             .findAll())
16         {
17             //sprawdzenie czy dana rezerwacja pokrywa się z
18             //przedziałem dat będącym parametrem tej funkcji
19             if(reservation.getRoom().getId() ==
20                 roomType.getId() && ((startDate.after
21                     (reservation.getStartDate())
22                     && startDate.before(reservation.getEndDate())) ||
23                     (endDate.before(reservation.getEndDate()) &&
24                     endDate.after(reservation.getStartDate())) ||
25                     (startDate.before(reservation.getStartDate()) &&
26                     endDate.after(reservation.getStartDate()))))
27             {
28                 numberOfRoomsAvailable--;
29                 //jeżeli nie ma już wolnego pokoju, to program
30                 //go nie zwraca
31                 if(numberOfRoomsAvailable <= 0)
32                 {
33                     isAvailable = false;
34                     break;
35                 }
36             }
37         }
38         //jeżeli jest wolny pokój, to jest dodawany do listy
39         if(isAvailable)
40             roomsAvailable.add(roomType);
41     }
42 }
43
44 return roomsAvailable;
}
```

6. Weryfikacja i walidacja

W rozdziale tym opisany został sposób testowania w ramach pracy. Opisane zostały przypadki testowe oraz zakres testowania, a także wykryte i usunięte błędy.

6.1 Sposób testowania w ramach pracy

W trakcie przebiegu procesu implementacji aplikacji, system był na bieżąco testowany pod kątem poprawności działania za pomocą techniki modelu przyrostowego. W przypadku każdej modyfikacji programu aplikacja była ponownie uruchamiana i testowana. Wykorzystywano także konsolę środowiska programistycznego w przypadku serwera oraz konsolę przeglądarki w przypadku klienta w celu wykrycia ewentualnych błędów.

6.2 Organizacja eksperymentów

Dokonano kilku przykładowych eksperymentów. Podczas ich przebiegu przeprowadzono proces rejestracji nowego klienta, zalogowanie się do systemu, a także zarezerwowania przykładowego pokoju. Przetestowano także obsługę administracyjną systemu, między innymi dodawanie nowych parametrów pokoju oraz nowego rodzaju pokoju do systemu.

6.3 Wykryte i usunięte błędy

Podczas przebiegu procesu testowania zostały zauważone błędy w działaniu aplikacji. Zauważono problem z przesyłaniem danych pomiędzy klientem, a serwerem systemu, należało także poprawić wygląd klienta poprzez odpowiednie sformatowanie tabel z aktualnie używanymi parametrami pokoi oraz ich rodzajem przez administratora systemu. Wszystkie te błędy zostały usunięte

7. Podsumowanie i wnioski

Po zaimplementowaniu wszystkich komponentów systemu uzyskano jako wynik aplikację spełniającą zakładane wcześniej założenia funkcjonalne. Realizacja systemu jako całości była procesem czasochłonnym oraz skomplikowanym. Wymagała znajomości języków Java, TypeScript, HTML oraz CSS.

Warto podkreślić, że istnieje możliwość rozbudowy systemu poprzez dalsze dopracowanie widoku aplikacji lub rozszerzenie funkcjonalności programu. Jednym z proponowanych rozwiązań jest rozwinięcie panelu podglądu obłożenia poprzez uwzględnienie dokładnego terminu, w którym występują rezerwacje. Kolejną opcję stanowi dodanie możliwości udzielenia rabatu dla lojalnego klienta lub dodanie opcji rezerwacji grupowej. Możliwe jest także rozszerzenie programu o możliwość wysyłania maili do użytkownika w przypadku założenia konta w systemie oraz dokonania rezerwacji.

Bibliografia

- [1] Adam Freeman. *Angular. Profesjonalne techniki programowania. Wydanie II*. Wydawnictwo Helion, Gliwice, 2018.
- [2] Craig Walls. *Spring w Akcji. Wydanie V*. Wydawnictwo Helion, Gliwice, 2020.
- [3] Jeremy Wilken. *Angular w Akcji*. Wydawnictwo Helion, Gliwice, 2019.
- [4] *Hotel explorer- hotelarstwo, historia hotelarstwa na świecie - Dawno, dawno temu...*
https://www.hotelexplorer.pl/2,historia_hotelarstwa_na_swiecie_dawno_dawno_temu [data dostępu: 2020-12-19].
- [5] *Wikipedia - hotelarstwo*
<https://pl.wikipedia.org/wiki/Hotelarstwo> [data dostępu: 2020-12-19].
- [6] *Ustawa z dnia 29 sierpnia 1997 r. o usługach turystycznych.*
<https://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU19971330884/O/D19970884.pdf> [data dostępu: 2020-12-19].

- [7] *Rozporządzenie Ministra Gospodarki i Pracy z dnia 19 sierpnia 2004 roku, w sprawie obiektów hotelarskich i innych obiektów, w których świadczone są usługi hotelarskie.*
<http://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU20041881945/O/D20041945.pdf> [data dostępu: 2020-12-19].
- [8] *Hotel Memorial - Standardy hoteli na świecie.*
<http://www.hotel-memorial.cz/standardy-hoteli-na-swiecie/>
[data dostępu: 2020-12-19].
- [9] *Usługi towarzyszące w hotelach - bezpłatne i luksusowe.*
<https://intelihotel.pl/uslugi-towarzyszace-w-hotelach-bezplatne-i-luksusowe/> [data dostępu: 2020-12-19].
- [10] *Poradnik urlopowy - Gdzie i kiedy jechać na urlop? Co zabrać, na co zwracać uwagę i wiele innych porad - Doba hotelowa – od której do której godziny?*
<https://poradnikurlopowy.pl/doba-hotelowa-od-ktorej-do-ktorej-godziny/> [data dostępu: 2020-12-19].
- [11] *Hotelarstwo - Do czego jest potrzebna struktura organizacyjna hotelu? Dobry schemat organizacyjny to połowa sukcesu.*
<http://www.hotelarze.pl/hot/schemat-org.html> [data dostępu: 2020-12-22].
- [12] *Rodzaje rezerwacji hotelowych.*
<http://pl.hotelopedia.org/rodzaje-rezerwacji-hotelowych/>
[data dostępu: 2020-12-23].

-
- [13] *Typy pokoi hotelowych.*
<https://www.esky.pl/porady-dla-podroznych/hotele/rodzaje-hotelowych-pokoi/typy-pokoi-hotelowych> [data dostępu: 2020-12-23].
- [14] *Principled Design of the Modern Web Architecture*
<https://www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf> [data dostępu: 2021-01-12].
- [15] *Richardson Maturity Model*
<https://martinfowler.com/articles/richardsonMaturityModel.html> [data dostępu: 2021-01-12].
- [16] *Hibernate*
<https://vavatech.pl/technologie/frameworki/hibernate> [data dostępu: 2021-01-12].

Spis skrótów i symboli

<i>MVC</i>	model – widok – kontroler (ang. <i>model–view–controller</i>)
<i>SGL</i>	jedynka – pokój dla jednego gościa z jednym łóżkiem (ang. single room)
<i>DBL</i>	dwójka – pokój przeznaczony dla dwóch osób, posiadający jedno dwuosobowe łóżko, zwane „małżeńskim” (ang. double room) .
<i>CRS</i>	komputerowe systemy rezerwacji – (ang. Computerised Reservation Systems).
<i>UML</i>	zunifikowany język modelowania – (ang. Unified Modeling Language)
<i>API</i>	interfejs programowania aplikacji – (ang. application programming interface)
<i>HTML</i>	hipertekstowy język znaczników – (ang. HyperText Markup Language)
<i>REST</i>	styl architektury oprogramowania wykorzystywany przy tworzeniu aplikacji internetowych - (ang. Representational state transfer)
<i>JPA</i>	oficjalny standard mapowania obiektowo-relacyjnego dla języka programowania Java - (ang. Java Persistence API)
<i>CRUD</i>	stwórz, odczytaj, zaktualizuj, usuń - (ang. create , read, update, delete)
<i>JSF</i>	szkielet upraszczający tworzenie interfejsu użytkownika aplikacji – (ang. Java Server Faces)
<i>EE</i>	(ang. Enterprise Edition)
<i>SPA</i>	jednostronicowa aplikacja internetowa - posiadającą tylko jeden plik html – (ang. Single Page Application)
<i>CLI</i>	wiersz poleceń - (ang. command line interpreter/interface)

<i>CSS</i>	język służący do opisu formy prezentacji (wyświetlania) stron WWW – (ang. Cascading Style Sheets)
<i>JRE</i>	środowisko uruchomieniowe Javy – (ang. Java Runtime Environment)
<i>JWT</i>	(ang. JSON Web Token)
<i>HTTP</i>	protokół przesyłania dokumentów hipertekstowych - (ang. Hypertext Transfer Protocol)
<i>JSON</i>	lekki format wymiany danych komputerowych – (ang. JavaScript Object Notation)
<i>ORM</i>	Mapowanie obiektowo-relacyjne (ORM – ang. Object-Relational Mapping).

Zawartość dołączonej płyty

Na płycie DVD dołączonej do dokumentacji znajdują się następujące materiały:

- praca w formacie pdf,
- źródła programu,

Spis rysunków

Rys. 3.1 - Diagram przypadków użycia systemu.

Rys. 4.1 - Ekran widoczny po uruchomieniu aplikacji.

Rys. 4.2 - Ekran widoczny po próbie rezerwacji przez niezarejestrowanego użytkownika.

Rys. 4.3 - Ekran widoczny po zalogowaniu się jako administrator.

Rys. 4.4 - Ekran widoczny po przejściu do panelu edycji parametrów.

Rys. 4.5 - Ekran widoczny po wciśnięciu przycisku „edytuj parametry”.

Rys. 4.6 - Ekran widoczny po wciśnięciu przycisku „kontynuuj”.

Rys. 4.7 - Ekran widoczny po wciśnięciu przycisku „przejdź do podglądu twoich rezerwacji”.

Rys. 4.8 - Ekran widoczny po wciśnięciu przycisku „zarezerwuj pokój”.

Rys. 4.9 - Ekran widoczny po wciśnięciu przycisku „przejdź dalej”.

Rys. 4.10 - Ekran widoczny po wciśnięciu przycisku „zarezerwuj pokój”.

Rys. 5.1 - Zrzut ekranu przedstawiający schemat bazy danych systemu.

Rys. 5.2 - Przykładowy zrzut ekranu przedstawiający widok panelu administratora pgAdmin4.

Rys. 5.2 - Zrzut ekranu przedstawiający strukturę katalogów w serwerze aplikacji.

Rys. 5.3 - Zrzut ekranu przedstawiający strukturę katalogów w kliencie aplikacji.

Spis tabel

Listing 5.1 - Funkcja zwracająca wolne pokoje w danym okresie czasu.