

Zastosowanie Informatyki w Gospodarcie – Projekt
Temat: System wspomagania Hotelu

Zespół projektowy:

- Marek Gackowski – kierownik programistów
- Rafał Hejna – programista back-end
- Mateusz Kawa – kierownik projektu
- Paweł Stanek – programista front-end
- Łukasz Mendak – programista bazy danych

Spis treści

1 Wstęp.....	3
2 Wymagania Funkcjonalne.....	3
3 Wymagania pozafunkcjonalne.....	6
4 Budżet.....	7
5 Diagram Gantt.....	7
6 Analiza ryzyka.....	8
7 Technologia.....	9
8 Zarys Architektury.....	9

1 Wstęp

Celem projektu jest stworzenie witryny komputerowego wsparcia hotelu. Poprzez stronę można zapoznać się z ofertą, dokonywać rezerwacji, opłat. Witryna ma być przede wszystkim wsparciem dla osoby pracującej na recepcji. Dodatkowo ma wspierać pracę osobę z księgowości, a ostatecznie służy również dla klientów w celu rezerwacji pokoi czy też dokonywania opłat.

2 Wymagania Funkcjonalne

<i>Rejestracja użytkownika</i>	
ID:	rej_uz
Opis:	Strona umożliwia rejestrację użytkownika po wprowadzeniu odpowiednich danych (login, hasło, imię, nazwisko).
Priorytet:	Wysoki
Powiązania:	-

<i>Logowanie użytkownika</i>	
ID:	log_uz
Opis:	Strona umożliwia logowanie użytkownika po wprowadzeniu odpowiednich danych (login, hasło). Aplikacja powinna potwierdzić dane za pomocą bazy danych.
Priorytet:	Wysoki
Powiązania:	-

<i>Zabezpieczenie formularza przed botami</i>	
ID:	zab_bot
Opis:	System formularzy powinien zostać zabezpieczony przed botami. Może generować proste kody weryfikacyjne
Priorytet:	Niski
Powiązania:	rej_uz, log_uz

<i>Sprawdzenie dostępnych pokoi</i>	
ID:	spr_dos_pok
Opis:	Strona umożliwia przegląd stanu pokoi: wylistowanie ich, stan aktualny (wolny/zajęty/zarezerwowany).
Priorytet:	Średni
Powiązania:	-

<i>Rezerwacja pokoju</i>	
ID:	rez_pok
Opis:	Powinna być możliwość rezerwacji wybranego pokoju.
Priorytet:	Wysoki
Powiązania:	spr_dos_pok

<i>Wylistowanie wolnych pokoi</i>	
ID:	wyl_wol_pok
Opis:	Wylistowanie tylko wolnych pokoi w danym momencie.
Priorytet:	Wysoki
Powiązania:	spr_dos_pok

<i>Sprawdzenie nachodzenia na siebie rezerwacji</i>	
ID:	spr_nak_rez
Opis:	System powinien nie zezwolić na rezerwację, której górna granica nachodzi na dolną innej w przypadku tego samego pokoju.
Priorytet:	Średni
Powiązania:	rez_pok, zm_rez

<i>Sprawdzenie danych klienta</i>	
ID:	spr_dan_kli
Opis:	Powinna być możliwość wylistowania danych klienta, który dokonał rezerwacji przy posiadaniu pewnych uprawnień.
Priorytet:	Średni
Powiązania:	rez_pok

<i>Stworzenie klienta</i>	
ID:	stw_kli
Opis:	Możliwość stworzenia profilu klienta bez zakładania konta dla użytkowników uprawnionych.
Priorytet:	Średni
Powiązania:	-

<i>Usunięcie rezerwacji</i>	
ID:	us_rez
Opis:	Usunięcie rezerwacji powinno być dostępne dla klienta, który dokonał rezerwacji oraz dla użytkowników uprawnionych.
Priorytet:	Wysoki
Powiązania:	rez_pok

<i>Zmiana rezerwacji</i>	
ID:	zm_rez
Opis:	Zmiana rezerwacji powinna być dostępna dla klienta, który dokonał rezerwacji oraz dla użytkowników uprawnionych.
Priorytet:	Wysoki
Powiązania:	rez_pok

<i>Zmiana danych konta</i>	
ID:	zm_dan_kli
Opis:	Użytkownik oraz użytkownik z uprawnieniami powinien mieć możliwość zmodyfikować dane osobowe w systemie (imie i nazwisko).
Priorytet:	Niski
Powiązania:	rej_uz

<i>Zapłata</i>	
ID:	zap
Opis:	Użytkownik, który dokonał rezerwacji oraz użytkownik uprawniony powinien mieć możliwość uiścić opłatę (poprzez naciśnięcie przycisku)
Priorytet:	Średni
Powiązania:	rez_pok

<i>Sprawdzenie końca rezerwacji</i>	
ID:	spr_kon_rez
Opis:	Użytkownik powinien mieć możliwość sprawdzić kiedy dany pokój będzie dostępny (jeżeli jest aktualnie w statusie zarezerwowany lub zajęty).
Priorytet:	Średni
Powiązania:	spr_dos_pok

3 Wymagania pozafunkcjonalne

<i>Interface</i>	
ID:	interface
Opis:	Dostęp do systemu odbywać się będzie za pomocą przeglądarki internetowej.
Priorytet:	Wysoki
Powiązania:	-

<i>Wspierane systemy operacyjne</i>	
ID:	wsp_sys_op
Opis:	System powinien działać na przeglądarkach chrome, firefox oraz systemach operacyjnych wspierających i obsługujących html, css, js.
Priorytet:	Wysoki
Powiązania:	-

4 Budżet

Koszty Pracowników:

<i>Nazwa</i>	<i>Koszt</i>	<i>Waluta</i>
Kierownik projektu 45h x 90 EUR/h	4050	EUR
Kierownik Programistów 60h x 70 EUR/h	4200	EUR
3 x Programista 60h x 50 EUR/h	9000	EUR
Suma	17250	EUR

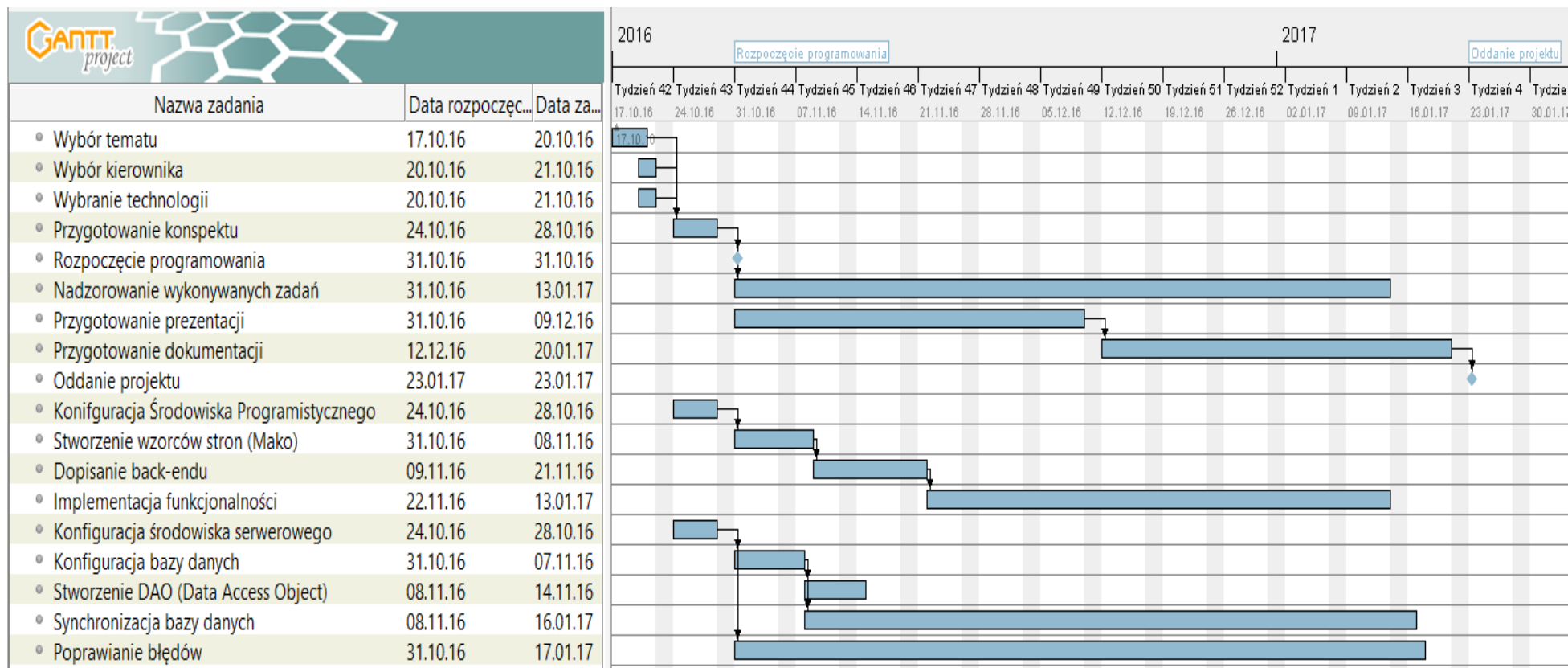
Dodatkowe:

Ubezpieczenie (19,64%)	3356,85	EUR
Koszty pośrednie (20%)	4121,37	EUR

Koszt projektu:

Suma Kosztów	24728,22	EUR
Zysk	0	EUR
VAT (19%)	4698,36	EUR
Koszt projektu	29426,58	EUR

5 Diagram Gantta



6 Analiza ryzyka

<i>Podział zadań w zespole</i>	
Prawdopodobieństwo:	30%
Wpływ:	Opóźnienie realizacji projektu

<i>Niedoszacowanie kosztów</i>	
Prawdopodobieństwo:	5%
Wpływ:	Zmniejszenie wynagrodzeń, redukcja etatów

<i>Błędy harmonogramu</i>	
Prawdopodobieństwo:	2%
Wpływ:	Opóźnienie realizacji projektu

<i>Modyfikacje członków zespołu</i>	
Prawdopodobieństwo:	13%
Wpływ:	Opóźnienie realizacji projektu

<i>Dodatkowe wymagania funkcjonalne klienta</i>	
Prawdopodobieństwo:	95%
Wpływ:	Zwiększenie kosztów projektu

<i>Nieznajomość używanych technologii</i>	
Prawdopodobieństwo:	13%
Wpływ:	Opóźnienie realizacji projektów, dodatkowe szkolenia

7 Technologia

Back-end	Python 3.5.2
Back-end serwer	Tornado (Python Module)
Front-end	HTML, CSS, JS
Front-end <-> Back-end	Mako (Python Template Module)
Serwer	BSD
Baza Danych	PostgreSQL 9.6

8 Zarys Architektury

