**Dokumentacja projektu zaliczeniowego**

Przedmiot: Inżynieria oprogramowania

Temat: **Biuro podróży**

Autorzy: Bogua Aleksander, Jadwiżyc Patryk, Kalisz Krzysztof

Grupa: 20B

Kierunek: Informatyka

Rok akademicki: 23/2024

Poziom i semestr: II/2

Tryb studiów: stacjonarne/niestacjonarne

*Należy pozostawić wszelkie nagłówki tego dokumentu, a umieszczać treść w odpowiednich miejscach zamiast obecnych objaśnień.*

*Stronę tytułową można sformatować w dowolny sposób, ale należy pozostawić zawartość informacyjną w układzie pokazanym powyżej.*

*Praca powinna zostać złożona wyłącznie w formacie pdf. Przed wygenerowaniem ostatecznej wersji należy zaktualizować spis treści – wyświetlane dwa poziomy.*

*Niniejszą informację należy również usunąć z wersji końcowej.*

* **Odnośniki do innych źródeł**

tj. do wykorzystywanych narzędzi / projektów w tych narzędziach

* Zarządzania projektem – Jira
* Wersjonowanie kodu – rekomendowany rzez lekarza farmaceutę i dwóxh fizjoterapeutów n a krzyż Git
* System obsługi defektów – Github
* **Słownik pojęć**

Tabela lub lista z pojęciami, które wymagają wyjaśnienia, wraz z tymi wyjaśnieniami – w szczególności synonimy różnych pojęć używanych w dokumentacji.

* **Wprowadzenie**
* ***Cel dokumentacji***

Celem niniejszej dokumentacji jest uchwycenie

* ***Przeznaczenie dokumentacji***

dla kogo ona jest

* ***Opis organizacji lub analiza rynku***

Jedna z dwóch opcji:

* Jeśli dla konkretnej organizacji: Czym jest organizacja, dla której realizowany będzie system; jak działa (lub będzie działała), kiedy system będzie wdrożony – tutaj nie odwołujemy się do samego systemu, tylko opisujemy samo działanie organizacji i role. W szczególności – jak wyglądają główne procesy biznesowe.
* Jeśli na masowy rynek: Pobieżna analiza rynku. Dla kogo będzie przydatny taki system. Ile jest organizacji, które będą mogły z niego skorzystać, użytkowników w poszczególnych organizacjach. Czy te organizacje stanowią jednorodną grupę czy są różne rodzaje. Co one mają ze sobą wspólnego. Jak ta liczba będzie się zmieniała w najbliższej przyszłości.
* Aplikacja przeznaczona jest na rynek masowy, celem jej bedzie agregowanie biur podrózy, klientów i podwykonawców, którzy będzą mogli łączyć się w ramach standaryzowanych umów. Niskie zagrożenia ze względu na klauzulę zrzeczenia się odpowiedzialności pośrednika. (4,5k biur podróży, 54% - 64% polaków jeżdżących na wakacje co roku) Aplikacja może być również znakomita okazją dla podwykonawców mających problmy z przedstawieniem swoich ofert dużym biurom podrży i mających problem z dotarciem do małych rozsiasnych po polsce interresów rodiznnych.
* ***Analiza SWOT organizacji***
* ***w polsce mamy 4,5 k biur podróży***
* jeśli system dla konkretnej organizacji:
* wystarczy sama tabela 2x2 (silne-słabe-szanse-zagrożenia)
* jeśli system na masowy rynek:
* szanse i zagrożenia
* **Specyfikacja wymagań**
* ***Charakterystyka ogólna***
* **Definicja produktu**

jedno zdanie o systemie – nazwa i rodzaj

* **Podstawowe założenia**

do czego będzie służył ten system – kilka/kilkanaście zdań wprowadzających

* **Cel biznesowy**

co organizacja docelowa chce osiągnąć wdrażając system

* **Użytkownicy**

lista – ew. wyjaśnienia dodać do słownika pojęć

* **Korzyści z systemu**

dla poszczególnych grup użytkowników – każdy element z unikalnym numerem identyfikacyjnym

* **Ograniczenia projektowe i wdrożeniowe**

przepisy prawne, specyficzne technologie, narzędzia, b.d., protokoły komunikacyjne, aspekty zabezpieczeń, zgodność ze standardami, powiązania z innymi aplikacjami, platforma sprzętowa, system operacyjny, inne komponenty niezbędne do współpracy – wszystko wraz z uzasadnieniem!

* ***Wymagania funkcjonalne***
* **Lista wymagań**

lista numerowana – czyli lista przypadków użycia lub bardziej ogólnie sformułowane wymagania, np. wymagania użytkownika

* **Diagramy przypadków użycia**

Tutaj same diagramy – bez specyfikacji, ale każdy diagram z tytułem i na osobnej stronie

* **Szczegółowy opis wymagań**

dla 5-7 wybranych najważniejszych przypadków użycia – przypadku zespołów 3-osobowych i większych, proporcjonalnie więcej

każde na nowej stronie wg następujących punktów:

* Numer – jako ID
* Nazwa
* Uzasadnienie biznesowe – odwołanie (-a) do elementów wymienionych w 5.1.5. (id i treść elementu, do którego się odwołujemy)
* Użytkownicy
* Scenariusze, dla każdego z nich:
* Nazwa scenariusza
* Warunki początkowe
* **Przebieg działań –** numerowana lista kroków, ze wskazaniem, kto realizuje dany krok
* Efekty – warunki końcowe
* Wymagania niefunkcjonalne – szczegółowe wobec poszczególnych wymagań funkcjonalnych
* Częstotliwość - na skali 1-5 lub BN-BW
* Istotność – inaczej: zależność krytyczna, znaczenie - na skali 1-5 lub BN-BW

***Ważne!***

*Elementy od warunków początkowych do końca mogą być grupowane, tj. specyfikacja pojedynczego przypadku użycia może zawierać:*

*- pojedynczy przebieg działań (scenariusz główny) oraz ew. scenariusze alternatywne, albo*

*- wiele przebiegów głównych wraz z ew. scenariuszami alternatywnymi – wtedy każdy z przebiegów głównych powinien być opisany wg tych punktów (od warunków początkowych do końca).*

* ***Wymagania niefunkcjonalne***

W odniesieniu do całego systemu, modułów lub innych składowych systemu

* Wydajność – w odniesieniu do konkretnych sytuacji – funkcji systemu
* Bezpieczeństwo – utrata, zniszczenie danych, zniszczenie innego systemu przez nasz – wraz z działaniami zapobiegawczymi i ograniczającymi skutki
* Zabezpieczenia
* Inne cechy jakości – najlepiej ilościowo, żeby można było zweryfikować (zmierzyć) – adaptowalność, dostępność, poprawność, elastyczność, łatwość konserwacji, przenośność, awaryjność, testowalność, użyteczność
* **Zarządzanie projektem**
* ***Zasoby ludzkie***

(rzeczywiste lub hipotetyczne) – przy realizacji projektu

Należy założyć, że projekt byłby realizowany w całości jako projekt komercyjny a nie tylko częściowo w ramach zajęć na uczelni

* ***Harmonogram prac***

Etapy mogą się składać z zadań.

Wskazać czasy trwania poszczególnych etapów i zadań – wykres Gantta.

obejmuje również harmonogram wdrożenia projektu – np. szkolenie, rozruch, konfiguracja, serwis – może obejmować różne wydania (tj. o różnej funkcjonalności – personal, professional, enterprise) i wersje (1.0, 1.5, itd.)

* ***Etapy/kamienie milowe projektu***

dla głównych etapów projektu

W grupach prowadzonych przez ŁR ten cały ten rozdział jest opcjonalny – dla chętnych. Nie jest omawiany na wykładzie!

Studenci powinni skonsultować szczegółowe wymagania w tym zakresie z nauczycielem prowadzącym zajęcia w danej grupie.

* **Zarządzanie ryzykiem**
* ***Lista czynników ryzyka***

Wypełniona lista kontrolna

* ***Ocena ryzyka***

prawdopodobieństwo i wpływ

* ***Plan reakcji na ryzyko***

Działania w odniesieniu do poszczególnych ryzyk.

Mogą być wg różnych strategii, tj. kilka strategii dla pojedynczego czynnika ryzyka

Rozdział obowiązkowy w zespołach co najmniej 3-osobowych, w mniejszych – do uzgodnienia z prowadzącym zajęcia.

* **Zarządzanie jakością**
* ***Scenariusze i przypadki testowe***

głównie testowanie funkcjonalności, ale może być też testowanie wymagań niefunkcjonalnych/zgodności; każdy scenariusz od nowej strony, musi zawierać co najmniej następujące informacje (sugerowany układ tabelaryczny, np. wg szablonu podanego w osobnym pliku lub na wykładzie):

* numer – jako ID
* nazwa scenariusza – co test w nim testowane (max kilka wyrazów)
* kategoria – poziom/kategoria testów
* opis – dodatkowe opcjonalne informacje, które nie zmieściły się w nazwie
* tester - konkretna osoba lub klient/pracownik,
* termin – kiedy testowanie ma być przeprowadzane,
* narzędzia wspomagające – jeśli jakieś są używane przy danym scenariuszu
* przebieg działań – tabela z trzema kolumnami: lp. oraz opisującymi działania testera i systemu
* założenia, środowisko, warunki wstępne, dane wejściowe – przygotowanie przed uruchomieniem testów
* zestaw danych testowych – najlepiej w formie tabelarycznej – jakie konkretnie dane mają być użyte przez testera i zwrócone przez system w poszczególnych krokach przebiegu działań
* *przebieg lub zestaw danych testowych musi zawierać jawną informację o warunku zaliczenia testu*
* **Projekt techniczny**
* ***Opis architektury systemu***

z ew. rysunkami pomocniczymi

* ***Technologie implementacji systemu***

tabela z listą wykorzystanych technologii, każda z uzasadnieniem

* ***Diagramy UML***

każdy diagram ma mieć tytuł oraz ma być na osobnej stronie

diagramy przypadków użycia umieszczone w punkcie 5.2.2, a nie tutaj.

* **Diagram(-y) klas**

1 lub więcej

* **Diagram(-y) czynności**

Co najmniej 1 dla zespołów 2-osobowych, więcej dla liczniejszych

* **Diagramy sekwencji**

co najmniej 5, w tym co najmniej 1 przypadek użycia zilustrowany kilkoma diagramami (dla zespołów 2-osobowych, dla liczniejszych więcej)

* **Inne diagramy**

co najmniej trzy – komponentów, rozmieszczenia, maszyny stanowej itp.

* ***Charakterystyka zastosowanych wzorców projektowych***

informacja opisowa wspomagana diagramami (odsyłaczami do diagramów UML); jeśli wykorzystano wzorce projektowe, to należy wykazać dwa z nich

uwaga – wzorce projektowe nie są omawiane na wykładach!

* ***Projekt bazy danych***
* **Schemat**

w trzeciej formie normalnej; jeśli w innej to umieć uzasadnić wybór

* **Projekty szczegółowe tabel**

w zależności, czy następujące elementy są widoczne na schemacie b.d.: nazwa tabeli, nazwy pól, typ danych, wartości NULL, klucz główny, klucz obcy –

- jeśli TAK: i nie ma potrzeby pokazania dodatkowych elementów b.d., to ten punkt może być pusty,

- jeśli NIE: to podać te elementy, których nie widać na schemacie.

dodatkowymi elementami mogą być np. triggery, procedury, funkcje, indeksy, użytkownicy, role.

* ***Projekt interfejsu użytkownika***

co najmniej dla głównej funkcjonalności programu – w razie wątpliwości, uzgodnić z prowadzącym zajęcia

* **Lista głównych elementów interfejsu**

okien, stron, aktywności (Android)

* **Przejścia między głównymi elementami**

np. storyboard, schemat blokowy lub inna notacja

* **Projekty szczegółowe poszczególnych elementów**

dla 5-7 głównych elementów (w zespołach 2-osobowych)

każdy element od nowej strony z następującą minimalną zawartością:

* numer – ID elementu
* nazwa – np. formularz danych produktu
* projekt graficzny – wystarczy schemat w narzędziu graficznym lub zrzut ekranu – z przykładowymi danymi (nie pusty!!!)
* opcjonalnie:
* opis – dodatkowe opcjonalne informacje o przeznaczeniu, obsłudze – jeśli nazwa nie będzie wystarczająco czytelna
* wykorzystane dane – jakie dane z bazy danych są wykorzystywane
* opis działania – tabela pokazująca m.in. co się dzieje po kliknięciu przycisku, wybraniu opcji z menu itp.
* ***Procedura wdrożenia***

jeśli informacje w harmonogramie nie są wystarczające (a zapewne nie są)

* **Dokumentacja dla użytkownika**

Opcjonalnie – dla chętnych

Na podstawie projektu docelowej aplikacji, a nie zaimplementowanego prototypu architektury

4-6 stron z obrazkami (np. zrzuty ekranowe, polecenia do wpisania na konsoli, itp.)

* pisana językiem odpowiednim do grupy odbiorców – czyli najczęściej nie do informatyków
* może to być przebieg krok po kroku obsługi jednej głównej funkcji systemu, kilku mniejszych, instrukcja instalacji lub innej pomocniczej czynności.
* **Podsumowanie**
* ***Szczegółowe nakłady projektowe członków zespołu***

tabela (kolumny to osoby, wiersze to działania) pokazująca, kto ile czasu poświęcił na projekt oraz procentowy udział każdej osoby w danym zadaniu oraz wiersz podsumowania – procentowy udział każdej osoby w skali całego projektu

* **Inne informacje**

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach