

Praca inżynierska

Jak pisać prace inżynierskie na kierunku Informatyka

Studia inżynierskie w Zachodniopomorskiej Szkole Biznesu kończą się napisaniem przez studenta pracy inżynierskiej oraz zdaniem egzaminu dyplomowego. W opracowaniu dowiesz się, jak określić temat i zakres pracy inżynierskiej oraz w jaki sposób ją napisać, aby odpowiadała wymogom.

REGULAMINU STUDIÓW ZACHODNIOPOMORSKIEJ SZKOŁY BIZNESU W SZCZECINIE

Pkt 14. § 32.

1. Procedura dyplomowa studiów pierwszego stopnia obejmuje egzamin dyplomowy oraz przygotowanie pracy dyplomowej (projektu końcowego).
2. Student przygotowuje projekt końcowy pod kierunkiem nauczyciela akademickiego posiadającego co najmniej stopień naukowy doktora. Możliwe jest wyznaczenie promotora niespełniającego tego warunku za zgodą Rady Wydziału.
3. Projekt końcowy powinien charakteryzować się pragmatycznym podejściem do rozwiązania konkretnego problemu. Może mieć charakter projektu analitycznego lub wdrożeniowego. Projekt końcowy musi obejmować komponent badań własnych studenta.
4. Projekt końcowy może być pracą zespołową przygotowaną maksymalnie przez trzy osoby. W przypadku opracowania projektu końcowego przez zespół studentów należy określić wkład każdego z członków zespołu w jego przygotowanie.
5. Projekt końcowy oceniany jest przez promotora i recenzenta.
6. Student zobowiązany jest złożyć w dziekanacie projekt końcowy w dwóch egzemplarzach oraz tekst projektu utwarty na płycie CD wraz z kartą obiegową najpóźniej na 14 dni przed planowanym terminem egzaminu

Na kierunku Informatyka w ramach pracy dyplomowej powinna powstać działająca aplikacja komputerowa lub mobilna, będąca rozwiązaniem problemu informatycznego.

Spis treści

OKREŚLANIE CELU I TEMATU PRACY	4
OKREŚLANIE ZAKRESU PRACY DYPLOMOWEJ	6
Zbyt wąski zakres pracy.....	6
Zbyt szeroki zakres pracy.....	7
STRUKTURA PRACY INŻYNIERSKIEJ.....	9
Wstęp.....	10
I Rozdział.....	11
II Rozdział.....	13
1. Cel i założenia projektu	13
2. Funkcjonalność projektu.....	14
3. Model danych systemu	15
4. Moduły aplikacji/Opis działania aplikacji.....	15
5. Wykorzystane technologie i narzędzia	16
6. Interfejs systemu	17
7. Testy działania.....	17
8. Opis wdrożenia.....	18
Podsumowanie i wnioski	18
Bibliografia.....	19
Spis rysunków, tabel, wykresów, załączniki.....	20
FORMATOWANIE PRACY	21
Marginesy	21
Formatowanie tekstu	21
Nagłówek i stopka	22
Wstawianie rysunków, wykresów, tabel, kodu	22
Tytuły rozdziałów	22
Spis treści	23

Określanie celu i tematu pracy

Dobranie właściwego tematu pracy nie jest rzeczą prostą. Przede wszystkim musimy mieć na względzie, że praca inżynierska to praca praktyczna, w rezultacie której powinna powstać działająca aplikacja komputerowa lub mobilna.

Rozsądnie jest najpierw określić, co chcemy zrobić, potem w jaki sposób chcemy to osiągnąć (przy pomocy jakich technologii, narzędzi), a dopiero na końcu sformułować temat pracy.

Musimy przy tym pamiętać, że nie tyle chodzi o to, czy zrobimy wielki portal, skomplikowaną aplikację mobilną, rozbudowane oprogramowanie w języku JAVA czy ASP, ale o to jaki problem informatyczny rozwiążemy, jakie technologie wykorzystamy i jak je ze sobą połączymy.

Celem pracy nie tyle jest wytworzenie konkretnego oprogramowania, co rozwiązanie problemu informatycznego.

Przykładowo:

Źle sformułowany temat: Projekt witryny internetowej do rysowania diagramów UML

Dobrze sformułowany temat: Wykorzystanie technologii SVG i obiektu CANVAS języka HTML5 do tworzenia diagramów UML na witrynie internetowej.

Jeśli rozwiązujesz jakiś problem informatyczny w pracy, możesz to oczywiście wykorzystać. Rozwiązania oparte o rzeczywiste problemy to doskonałe zagadnienie na pracę inżynierską.

Podczas określania tematu pracy możesz zasięgnąć rady u promotora. Możesz także zaproponować własny temat odnoszący się do dziedziny, która Cię interesuje.

Przykład

Jako przykład weźmy budowę serwisu internetowego. Studenci często przychodzą z taką propozycją tematu pracy dyplomowej. Właściwie zmienia się tylko tematyka:

- Projekt serwisu internetowego o Szczecinie
- Projekt serwisu internetowego dla hodowców psów
- Projekt serwisu internetowego o imprezach w mieście

Ten temat nic nie mówi – nie ma w nim słowa o problemie informatycznym, który ta praca ma rozwiązać, nie ma wzmianki o technologiach czy bibliotekach, które zostaną wykorzystane.

Tak naprawdę, nie ma znaczenia, czy w rezultacie powstanie coś o Szczecinie, psach czy imprezach. Ważne jest, w jaki sposób to zrealizujemy.

Jak więc mogłyby brzmieć te tematy?

- Projekt serwisu internetowego opartego o środowisko programistyczne node.js
- Projekt serwisu internetowego wykorzystującego AJAX do generowania podstron
- Serwis internetowy działający na rozproszonych bazach danych

W powyższych tematach jest wszystko, co potrzeba – problem informatyczny i technologia z nim związana.

- W jaki sposób napisać serwis na node.js?
- W jaki sposób wygenerować strony wykorzystując AJAX?
- W jaki sposób połączyć dane z rozproszonych baz danych?

Określanie zakresu pracy dyplomowej

Podobnie jak z określaniem celu i tematu pracy dyplomowej, również dobre przemyślenie zakresu tej pracy może zaważyć na tym, czy projekt zakończy się sukcesem, czy jedynie się zakończy.

Istnieją dwa zasadnicze błędy, które popełniają osoby rozpoczynające pisanie pracy inżynierskiej:

1. Zbyt wąski zakres pracy
2. Zbyt szeroki zakres pracy

Zbyt wąski zakres pracy

Skoro i tak musisz napisać pracę inżynierską, wykorzystaj ją dla własnego rozwoju. Wybierz technologię, której jeszcze nie znasz. Zastosuj rozwiązanie, które przyda Ci się w pracy.

Bardzo często podczas rozpoznawania technologii, które mają pomóc w realizacji pracy dyplomowej, student zawęża swoje wybory tylko do tego, co dobrze już zna. Prowadzi to do sytuacji, gdy wiele z możliwości zostaje odciętych automatycznie, już na samym początku, tylko dlatego, że jeszcze się z nimi nie spotkaliśmy.

To błąd – informatyka to dziedzina, która zmusza do ciągłej nauki i rozwoju, nie tylko na studiach, ale przez cały czas.

Zastanawiając się nad wykorzystaniem jakiejś technologii nie patrz na nazwę i swoje obecne umiejętności w jej zakresie. Sprawdź jej możliwości, dostępną dokumentację, fora z nią związane. Czasem niewielkim nakładem pracy można nauczyć się czegoś nowego i tym samym zwiększyć swoją wartość na rynku pracy.

Planuj jednak rozsądnie – lepiej ograniczyć się do jednej technologii i dobrze ją opanować, niż zakopać się w całej masie nowości, które nas w końcu przytłoczą i

sprawia, że nawet nie zaczniemy. Jedna – dwie nowe technologie na pracę inżynierską wystarczą.

Zbyt szeroki zakres pracy

Drugim problemem podczas określania zakresu projektu dyplomowego jest nadmierne rozbudowywanie projektu o funkcjonalności, które nic do niego nie wnoszą i ambicja, być zrobić więcej, niż jesteśmy w stanie.

Skupmy się na tym, co jest istotą problemu pracy. Na przykład jeśli uznaliśmy, że będziemy eksplorować możliwości jakiejś biblioteki, spróbujmy wykorzystać je do maksimum, bez dodawania innych, niepotrzebnych elementów zupełnie z nią nie związanych.

Jeśli tematem pracy jest wykorzystanie biblioteki do obróbki zdjęć, nie ma sensu dobudowywać do tego systemu zarządzania użytkownikami oraz modułu do przesyłania wiadomości. Lepiej dodać kolejną funkcjonalność do samego mechanizmu obrabiania zdjęcia, np.: wymyślić i zastosować dodatkowy filtr, poprawić jakość zapisywanych obrazów.

Zbyt wiele funkcjonalności odwraca naszą uwagę od tego, co chcemy zrobić w pracy oraz powoduje, że cel tej pracy się rozmywa. Recenzent na pewno to zauważy.

To samo tyczy się doboru technologii, o które oprzemy aplikację dyplomową. Często zdarza się tak, że chcemy wykorzystać w pracy jak najwięcej nowości, o których przeczytaliśmy ostatnio. To bardzo ryzykowne. Wybierzmy jedną, maksymalnie dwie nowe technologie, co do których jesteśmy pewni, że je opanujemy na tyle, by skończyć projekt. I że będziemy w stanie połączyć je w naszym projekcie w sensowny sposób.

Skupmy się na tym, co jest istotą problemu pracy.

Dodatkowe funkcjonalności nie związane z rzeczywistym celem pracy odrywają nas od tego, co istotne i zabierają czas.

Przykład

Byłam kiedyś promotorem pracy inżynierskiej, której celem było opracowanie metody doboru graczy do grup treningowych poprzez stronę online. Gracze rejestrowali się na stronie, wprowadzali swoją punktację w danej grze, zaś system na tej podstawie proponował im zespoły, przeciw którymi powinni zagrać.

Studenci przez dłuższy czas nie kontaktowali się ze mną podczas pisania aplikacji i pracy. Kiedy w końcu się skontaktowali, z dumą przedstawili mi rezultaty.

Aplikacja miała postać prostej witryny internetowej z informacjami o grze oraz generatorem zespołów.

Praca pisemna natomiast miała około 150 stron. Większość zajęł opis gry, jej mechanik, środowiska fanów, imprez z nią związanych. Druga część obejmowała opis budowy strony w php, jak zostało zrealizowane logowanie, jak menu, w jaki sposób zbudowano formularze. Nigdzie, w całej pracy autorzy nie zamieścili opisu metody, w jaki sposób dobierali zespoły.

Dużo czasu zajęło mi wytłumaczenie studentom, że nikt nie będzie czytał o tym, jak powstała gra i jak się rozwijała, bo nie o tym jest praca inżynierska. Byli zawiedzeni, bo bardzo się napracowali.

Tak samo nie mogli zrozumieć, że nie ma sensu opisywać w tej konkretnej pracy, w jaki sposób strona łączy się z bazą danych, jak wygląda logowanie i jak wyglądają formularze w HTML. To podstawowe zagadnienia, które nic nie wnoszą.

To, co było sednem pracy, to mechanizm dobierania zawodników. Należało opisać i zilustrować, w jaki sposób na podstawie wprowadzonych danych system określa grupy graczy.

Na koniec naszych rozmów i poprawek praca miała niecałe 70 stron, na których zostały opisane istniejące metody doboru zawodników oraz sposób klasyfikacji wykorzystany w pracy.

Struktura pracy inżynierskiej

Struktura pracy inżynierskiej na kierunku Informatyka różni się znacząco od struktury prac na innych kierunkach. Przede wszystkim jest to praca praktyczna, projektowa, w wyniku której powstaje aplikacja komputerowa lub mobilna.

Cała praca powinna zamykać się w 50-70 stronach. Może się wydawać, że to dużo, jednakże jeśli weźmiemy pod uwagę, iż w te 50 stron wchodzi także diagramy UML, kawałki kodu oraz zrzuty ekranów, okazuje się, że wcale nie jest tak trudno osiągnąć wymaganą ilość.

Aby dobrze zobrazować problem zawarty w pracy dyplomowej, warto przestrzegać poniższego układu.

WSTĘP

I ROZDZIAŁ (rozdział teoretyczny)

1. Wprowadzenie do problematyki pracy, geneza powstania problemu
2. Opis środowiska, w jakim zidentyfikowano problem
3. Przegląd i porównanie istniejących rozwiązań

II ROZDZIAŁ (rozdział praktyczny)

1. Cel i główne założenia projektu
2. Funkcjonalność projektu
3. Model danych aplikacji
4. Moduły aplikacji / Opis działania aplikacji
5. Wykorzystane technologie i narzędzia
6. Interfejs systemu
7. Testy działania
8. Wdrożenie systemu

PODSUMOWANIE I WNIOSKI

BIBLIOGRAFIA

SPIS RYSUNKÓW

SPIS TABEL

ZAŁĄCZNIKI

Wstęp

Wstęp to tak naprawdę streszczenie pracy obejmujące około 2 stron. To pierwszy element pracy, który zainteresuje recenzenta. Powinien być więc opracowany wyjątkowo starannie i dawać dobre pojęcie o tym, na co natkniemy się w dalszej części pracy.

Tak więc w tej części powinien znaleźć się krótki opis problemu pracy, jej cel i podstawowe założenia, krótki opis rozwiązania, a także informacja, w której części pracy co się znajduje. To także miejsce na krótkie podsumowanie prac nad projektem oraz określenie, komu opracowane rozwiązanie może się przydać.

Pisząc wstęp postaraj się nie stosować list wypunktowanych. Tekst powinien być zawarty w akapitach, przynajmniej po jednym na każde wymienione poniżej zagadnienie.

To nie żart, że Wstęp zazwyczaj piszemy na końcu. To streszczenie pracy obejmujące około 2 stron i element pracy, który na pewno zainteresuje recenzenta.

Wstęp powinien zawierać:

- *Wprowadzenie do problemu informatycznego,*
- *Cel pracy i podstawowe założenia,*
- *Krótki opis zastosowanego rozwiązania,*
- *Określenie grupy docelowej, do której kierowane jest rozwiązanie,*
- *Informację, z ilu rozdziałów składa się praca i co zawierają,*
- *Krótkie podsumowanie prac nad projektem.*

I Rozdział

Rozdział pierwszy to rozdział poświęcony zapoznaniu czytelnika z problemem informatycznym, jaki chcemy rozwiązać. Z tego powodu powinien on zawierać skrócone wprowadzenie do problemu – jak doszło do tego, że taki problem zaistniał i z czym jest związany.

Następnie należy przybliżyć przynajmniej 3 rozwiązania, które są dostępne na rynku. Przeważnie obejmuje to krótki opis rozwiązania, zrzut przykładowego ekranu oraz wyszczególnienie funkcjonalności, jakie dane rozwiązanie oferuje.

Dobłą praktyką jest, aby na końcu rozdziału pierwszego zrobić zestawienie możliwości opisanych rozwiązań z wyszczególnieniem mocnych i słabych stron. Dzięki temu będzie można wskazać, czemu wykonana przez nas aplikacja jest lepsza lub równie dobra, jak aplikacje przykładowe.

Przykład

Jeśli celem pracy jest stworzenie oprogramowania, które wspierałoby pracę zespołów rozproszonych korzystających z urządzeń desktopowych i mobilnych, w rozdziale pierwszym pracy należałoby opisać:

Jak obecnie wygląda praca nad projektami, w których członkowie zespołów znajdują się w różnych lokalizacjach geograficznych;

Jakie problemy napotykają zespoły rozproszone podczas realizacji projektów;

Jakie koszty zmiany w realizacji projektów po rozwinięciu się technologii mobilnych (np. ciągły dostęp do Internetu, cloud computing, itd.).

Opis 3 aplikacji do zarządzania pracą zespołową, np. Trello, Evernote, Flow

Podsumowanie wad i zalet powyższych aplikacji

W przypadku, gdy w efekcie pracy inżynierskiej ma powstać gra komputerowa pisana w środowisku do tego dedykowanym, w rozdziale pierwszym należy skupić się na metodach tworzenia gier komputerowych, mechanikach gry oraz opisie środowiska do tworzenia gry.

Przykład

Jeśli celem pracy jest stworzenie gry komputerowej multiplayer w środowisku do tworzenia gier XNA GAME STUDIO, w rozdziale pierwszym należałoby:

- *Opisać zjawisko gier komputerowych;*
- *Opisać podstawowe mechaniki tworzenia gier;*
- *Zaprezentować 2-3 środowiska do tworzenia gier;*
- *Opisać środowisko XNA GAME STUDIO, w którym gra będzie tworzona z uwzględnieniem jego możliwości, sposobu tworzenia gry, itd.*

Nieco inaczej będzie wyglądała sytuacja w przypadku dużego i złożonego systemu, nad którym będzie pracowała więcej niż jedna osoba.

W takim wypadku rozdział pierwszy nadal powinien skupiać się na opisanu przyczyn powstania zagadnienia, jednakże w dalszej części należy skupić się na opisanu mechanik danej aplikacji oraz podziale na moduły, które później będą realizowane oddzielnie przez osoby piszące pracę.

Przykład

Jeśli celem pracy jest stworzenie dużego systemu usprawniającego na wielu polach pracę w firmie poligraficznej, rozdział pierwszy powinien zawierać:

- *Charakterystykę firmy;*
- *Problemy, jakie oprogramowanie ma rozwiązać;*
- *Charakterystykę dostępnych rozwiązań oraz przyczyny, dla których firma zdecydowała się na wybór własnego oprogramowania;*
- *Podział oprogramowania na części składowe.*

Tytuł rozdziału pierwszego powinien dobrze odnosić się do jego zawartości, tak aby osoba czytająca pracę od razu wiedziała, czego się spodziewać.

Rozdział pierwszy **powinien zajmować około 20-25%** pracy dyplomowej

II Rozdział

Rozdział drugi to sedno pracy inżynierskiej. To tu należy przedstawić wszystkie zagadnienia związane z tworzeniem aplikacji dyplomowej.

W typowych pracach inżynierskich rozdział ten zostanie podzielony na kilka podstawowych podrozdziałów opisanych poniżej:

1. Cel i główne założenia projektu
2. Funkcjonalność projektu
3. Model danych
4. Moduły/Części składowe projektu/Sposób działania
5. Wykorzystane technologie i narzędzia
6. Interfejs systemu
7. Testy działania
8. Wdrożenie systemu

1. Cel i założenia projektu

Pierwszy podrozdział to rozdział poświęcony określeniu, jaki jest cel pracy inżynierskiej. Na początku dobrą praktyką jest ponowne przybliżenie problemu informatycznego i przyczyn wyboru takiego właśnie tematu pracy, dzięki czemu rozdział II będzie naturalną kontynuacją rozdziału I. Nie powinno to jednak zająć więcej niż 1-2 akapity.

Następnie należy zaprezentować i opisać cel pracy. Cel pracy powinien być jasny i konkretny, zamknięty w jednym, wyróżnionym zdaniu. W dalszej części akapitu można go bardziej rozwinąć i wyjaśnić.

Dalsza część podrozdziału powinna prezentować założenia projektu i jego cele cząstkowe, czyli to wszystko, co chcemy osiągnąć dodatkowo. To mogą być założenia dotyczące bezpieczeństwa, użyteczności aplikacji, szybkości jej działania, dostępności rozwiązań, wykorzystania nowych trendów lub oprogramowania. Wszystko to, co jest związane z tworzoną aplikacją, ale nie zmieściło się w podstawowym celu.

Dobrze sformułowane założenia są przydatne przy pisaniu aplikacji i warto do nich co jakiś czas zajrzeć, gdyż pozwalają utrzymać się w ryzach i nie zbaczać z obranej ścieżki, co jest częstym zjawiskiem podczas pisania prac inżynierskich.

Przykład

Jeśli celem pracy jest stworzenie oprogramowania, które wspierałoby pracę zespołów rozproszonych korzystających z urządzeń desktopowych i mobilnych, założenia i cele cząstkowe mogą objąć:

- *Działanie systemu oparte o chmurę (powiązanie systemu z Dropbox);*
- *Zabezpieczenia na wypadek zerwania połączenia z siecią (tryb off-line);*
- *Optymalizację przesyłu danych na urządzenia mobilne.*
- *itd*

2. Funkcjonalność projektu

Wielu studentów myli funkcjonalność z opisem działania aplikacji. To nie to samo. Funkcjonalność (ang. utility) to zbiór czynności, jakie można wykonać przy pomocy danej aplikacji. Będzie to więc opis lub lista czynności, jakie może wykonać użytkownik.

Przy opisie funkcjonalności ważne jest zastanowienie się, jakie grupy użytkowników będą z niej korzystały i do jakich funkcji będą miały dostęp. Na pewno inny zakres możliwych czynności będzie miał administrator systemu, inny użytkownik zalogowany, a inny osoba będąca tylko gościem. W rozdziale należy podać ten podział i wyszczególnić funkcjonalności dla każdej grupy użytkowników.

Kolejnym ważnym elementem tego rozdziału są diagramy przypadków użycia UML odwzorowujące funkcjonalność grup użytkowników.

Funkcjonalność (ang. utility)
to zbiór czynności, jakie
można wykonać przy
pomocy danej aplikacji.

3. Model danych systemu

Ten rozdział poświęcony jest strukturze danych przechowywanych i wykorzystywanych w aplikacji projektowej. Należy w nim opisać, w jaki sposób przechowywane są dane – w bazie danych, w plikach XML lub JSON, czy w jeszcze jakiś inny sposób.

Opisując bazę danych należy wspomnieć, jaki to typ bazy, a następnie opisać jej strukturę: rodzaje tabel, powiązania między nimi, dodać schemat bazy danych. Ponadto należy opisać, w jaki sposób zamodelowano dane istotne z punktu widzenia aplikacji, np.: system punktów zawodników w grze czy ścieżki przejść w systemie personalizacji.

Jeśli aplikacja korzysta z danych w formatach tekstowych również należy określić, jaki to format, rozpisać strukturę danych np. dla JSON, można też zamieścić listingi danych specyficznych dla danego projektu.

Pamiętajmy, żeby podczas dobierania elementów, które w tej części chcemy zaprezentować, skupić się na tych danych, które są sednem pracy.

Nie ma sensu dawać listingu lub opisywać sposobu przechowywania danych do logowania (chyba, że tematem pracy jest usprawnienie systemu logowania), gdyż są to dane typowe dla większości systemów i nie wnoszą nic nowego.

4. Moduły aplikacji/Opis działania aplikacji

Moduły aplikacji to nic innego jak opis części, z których składa się aplikacja oraz opis działania poszczególnych modułów. To zbiory funkcjonalności logicznie powiązane ze sobą i realizujące pewien konkretny zakres działań.

Moduły powinny być dobierane w sposób rozważny. Rozbijanie aplikacji na wiele małych modułów utrudni czytającemu pracę ogarnięcie całości działania aplikacji.

Po co tworzyć oddzielne moduły do rejestracji, logowania, zmiany hasła i zarządzania ustawieniami, jeśli wszystkie te elementy można połączyć i opisać jako moduł zarządzania użytkownikami?

Przykład

Jeśli w wyniku pracy inżynierskiej powstałaby aplikacja do zarządzania czasem pracy pracowników, system prawdopodobnie składałby się z modułów:

- Modułu administracyjnego – do zarządzania użytkownikami, konfigurowania aplikacji
- Modułu pracownika – zawierającego pewnie jakiś mechanizm śledzenia czasu pracy, zapisywania zadań, składania sprawozdań
- Modułu raportującego – służącego do generowania zestawień czasu pracy, archiwizowania raportów

Opisując poszczególne moduły należy opisać ich sposób działania, ale także wszystkie zagadnienia związane ze sposobem realizacji ich funkcjonalności. To tu jest miejsce na rozważania nad doбором metody rozwiązania danego problemu, na zaprezentowanie listingów kodu z ważniejszymi elementami aplikacji.

Tu także można umieścić diagramy klas UML czy też diagramy sekwencji UML, obrazujące działanie aplikacji.

5. Wykorzystane technologie i narzędzia

To jeden z najprostszych rozdziałów, a jednak najczęściej jest źle formułowany. W tej części pracy należy opisać, z jakich technologii, języków programowania, bibliotek, frameworków i narzędzi korzystano podczas tworzenia aplikacji projektowej.

Najczęstszym błędem popełnianym przez studentów jest skupianie się na każdej technologii i szczegółowe opisywanie jej historii, możliwości, przykładów.

To nie są rzeczy, które interesują recenzenta pracy. Recenzent chce wiedzieć, do czego w pracy wykorzystano daną technologię, a nie jak skonfigurować całe środowisko programistyczne. Dlatego w pracy ograniczamy się jedynie do podania nazwy technologii i jej wersji oraz opisu, do czego została przez nas wykorzystana.

W przypadku, gdy sednem pracy jest wykorzystanie jakiejś biblioteki i wskazanie jej możliwości, dokładny opis biblioteki znajdzie się albo w Rozdziale I pracy albo w Opisie działania aplikacji.

6. Interfejs systemu

Ten rozdział pracy ma na celu pokazanie, jak wygląda aplikacja oraz uzasadnienie takiego jej wyglądu/układu. Podczas opisu interfejsu należy posłużyć się zasadami użyteczności Jacoba Nielsena oraz User Experience, które są kanonem w budowaniu interfejsów.

Heurystyki Jacoba Nielsena i Ralfa Molicha

1. Pokazuj status systemu.
2. Zachowaj zgodność pomiędzy systemem a rzeczywistością.
3. Daj użytkownikowi pełną kontrolę.
4. Trzymaj się standardów i zachowaj spójność.
5. Zapobiegaj błędom.
6. Pozwalaj wybierać zamiast zmuszać do pamiętania.
7. Zapewnij elastyczność i efektywność.
8. Dbaj o estetykę i umiar.
9. Zapewnij skuteczną obsługę błędów.
10. Zadbaj o pomoc i dokumentację.

Oprócz kilku zrzutów ekranu (np. strona główna, panel wprowadzania danych, rozwinięte menu), należy uzasadnić strukturę menu, dobór kolorystyki, czcionek, opisać czy i w jaki sposób dostosowano aplikację do różnych typów urządzeń. Należy przy tym uważać, żeby nie przesadzić z ilością zrzutów ekranów – 3 do 5 z pewnością wystarczą.

7. Testy działania

Ta część pracy będzie występowała jedynie w przypadku aplikacji, które opierają się o analizę danych lub zachowań użytkowników. Powinny tu zostać opisane rodzaje testów, jakim poddano działanie aplikacji oraz zamieszczone wyniki tych testów z odpowiednimi wnioskami.

Do takich testów może np. należeć trafność typowania właściwych wyników w systemie personalizacji lub wydajność aplikacji.

8. Opis wdrożenia

W tej części należy wskazać, czy tworzona aplikacja została gdzieś wdrożona i z jakim skutkiem.

To także część poświęcona opisowi wymagań systemowych oraz sposobom instalacji aplikacji.

Przykład

W przypadku, gdy w wyniku pracy inżynierskiej powstanie witryna internetowa do wymagań systemowych może należeć:

- Serwer Apache v.2.4
- PHP v. 5.5.x
- Baza MySQL v. 5.x
- protokół SSL

Na koniec warto również podać przybliżony koszt wdrożenia.

Podsumowanie i wnioski

Zawartość tego rozdziału podsumowuje już sama nazwa. W tej części należy podsumować prace nad projektem, opisać co się udało osiągnąć, a czego nie oraz czy wybrane narzędzia i technologie spełniły nasze oczekiwania.

To także część pracy, w której należy zastanowić się nad dalszymi możliwościami rozwoju projektu: dodaniem nowych funkcjonalności, zastosowaniem innych rozwiązań.

Jeżeli projekt został wdrożony i działa, w podsumowaniu warto o tym poinformować, podać ewentualne statystyki albo rezultaty działania stworzonej aplikacji.

Bibliografia

Każda praca dyplomowa powinna zawierać zbiór źródeł i publikacji, na których opierano się przy jej pisaniu. W przypadku prac inżynierskich na kierunku Informatyka bywa to o tyle trudne, że często efektem pracy jest zupełnie nowa aplikacja. Jak więc dobrać źródła?

Przed wszystkim należy dobrze skonstruować rozdział pierwszy, w którym przybliżone zostaje zagadnienie leżące u podstaw tematu pracy. Należy poszukać literatury tematycznej i tu ją zamieścić. Także przy prezentacji przykładowych rozwiązań należy zamieścić przypisy do ich dokumentacji lub dotyczących ich opracowań.

Jeśli w pracy wykorzystywane są różne technologie, rozwiązania, narzędzia - warto zamieścić odnośniki do ich dokumentacji.

Przy opisie interfejsu można posilkować się źródłami dotyczącymi User Experience oraz użyteczności.

Samo zamieszczenie literatury i źródeł w dziale Bibliografia nie wystarczy. Powinny one być zawsze oznaczone także przypisem dolnym¹, w jednym z poniższych formatów:

- T. Żyro, *Wstęp do politologii*. Warszawa: PWN, 2006, s. 14-17.
- Marek, *Wpływ zmiany kultury organizacyjnej na zsl w międzynarodowym przedsiębiorstwie*. Zarządzanie Zasobami Ludzkimi, 2006, nr 5, s. 24-27.
- K. Denek, *Cywilizacja informacyjna i edukacja medialna*. [w:] T. Lewowicki, B. Siemieniecki (red.), *Rola i miejsce technologii informacyjnej w okresie reform edukacyjnych w Polsce*. Toruń: Adam Marszałek, 2002, s. 25-37.
- Prawo: *Biblioteki szkolne i pedagogiczne*, [dostęp 05.11.07], <http://ebib.oss.wroc.pl/prawo/bibliot/bibszkip.php>.

Korzystając ze źródeł należy pamiętać, że pisanie pracy dyplomowej nie polega na kopiowaniu całych stron i oznaczaniu ich jako źródła. Praca powinna być pisana własnym językiem jedynie w oparciu o źródła. Oczywiście można stosować cytaty, ale powinny one się ograniczyć do konkretnych definicji, poglądów, wypowiedzi, a nie bloków tekstu.

Unikamy zamieszczania jako literatury:

- *Wszelkiego rodzaju nieautoryzowanych kursów internetowych*
- *Pobieżnych i skrótowych informacji z Wikipedii – często pod artykułem na Wikipedii znajduje się spis odnośników do źródeł – lepiej to na nich oprzeć pracę*

Bibliografia powinna być ułożona w kolejności alfabetycznej po nazwisku autora lub chronologicznie – od najnowszych publikacji do najstarszych.

Dobłą praktyką jest wydzielenie źródeł internetowych.

Spis rysunków, tabel, wykresów, załączniki

Te elementy powinny znaleźć się na końcu pracy i zawsze zaczynać do nowej strony.

W spisach obowiązuje kolejność, w jakiej elementy pojawiają się w pracy. Przy tytule każdego rysunku, tabeli, wykresu powinna być też podana strona, na której on występuje.

Formatowanie pracy

Praca inżynierska powinna zawierać minimum 50 stron. W tym:

- WSTĘP – ok. 2 stron
- I ROZDZIAŁ – ok. 25% pracy
- II ROZDZIAŁ – ok. 70% prac
 - Cel i założenia – min. 1 strona
 - Funkcjonalność – min. 2 strony
 - Model danych – min. 2 strony
 - Technologie – min. 1 strona
 - Interfejs – min. 2 strony
- PODSUMOWANIE – ok. 3 stron

Marginesy

Marginesy w pracy powinny mieć po 2,5 cm z każdej strony, ewentualnie margines z lewej strony może być o centymetr większy ze względu na oprawę pracy. Wtedy margines prawy powinien zostać zmniejszony do 1,5 cm.

Formatowanie tekstu

- czcionka Times New Roman;
- kolor czcionki: czarny;
- rozmiar 12 pkt;
- odstęp między wierszami 1,5 linii;
- każdy akapit zaczyna się od wcięcia;
- tekst akapitów wyrównany do obu krawędzi;
- nie używamy wielokrotnej spacji (poza przypadkami koniecznymi);
- nie stawiamy spacji przed znakiem: , . : , ? ! % „) }];
- nie wstawiamy spacji za znakiem: ({ [”;

Nagłówek i stopka

U góry pracy powinien znajdować się nagłówek z tytułem pracy wypisanym wersalikami (czcionka 12 lub 11 pkt) i oddzielony linią od treści strony.

Numeracja strony powinna zaczynać się od strony pierwszej (z pominięciem strony tytułowej), ale być widoczna dopiero od Wstępu.

Wstawianie rysunków, wykresów, tabel, kodu

Każdy wstawiony do pracy element powinien być podpisany w odpowiedni sposób. Dla rysunku, wykresu czy wstawionego kodu podpis powinien znajdować się pod spodem.

Rys. 2. Tytuł rysunku

Źródło: Wykres Gantta z zasobami w Excelu jako mapa cieplna, [dostęp: 2012-04-10], <http://skuteczneraporty.pl/blog/wykres-gantta-z-zasobami-w-excelu-jako-mapa-cieplna/>

Wyk. 2. Tytuł wykresu

Źródło: MacCaw A.: JavaScript. Aplikcje WWW, Wydawnictwo Helion, Gliwice 2012.

Listing 2. Tytuł listingu

Źródło: opracowanie własne

W przypadku tabel tytuł powinien znajdować się nad tabelą.

Tab. 2. Tytuł tabeli

Każdy zamieszczony element powinien być opisany przynajmniej jednym tekstem akapitu. Opis powinien dotyczyć tego, co widać na rysunku/wykresie.

Tytuły rozdziałów

Główne rozdziały rozpoczynać należy od nowej strony. W pozostałych przypadkach należy unikać pozostawiania dużych pustych przestrzeni pomiędzy częściami pracy.

Nie należy zbyt zagnieżdżać treści i dzielić jej na wiele małych podrozdziałów. Standardowo stosuje się jedynie 3 poziomy zagnieżdżania.

- pierwszy poziom bold 16 wersaliki;

- drugi poziom bold 14;
- trzeci poziom bold 12.

Spis treści

Poniżej znajduje się przykładowy układ spisu treści.

SPIS TREŚCI

WSTĘP.....	5
I. TYTUŁ PIERWSZEGO ROZDZIAŁU.....	6
1. Tytuł pierwszego podrozdziału.....	6
2. Tytuł drugiego podrozdziału.....	8
2.1. Tytuł działu.....	8
2.2. Tytuł drugiego działu.....	9
II. PROJEKT SYSTEMU... (TYTUŁ PRACY).....	10
1. Cel i założenia projektu.....	10
2. Funkcjonalność projektu.....	10
3. Model danych.....	11
4. Moduły systemu.....	11
5. Wykorzystane narzędzia i technologie.....	11
6. Interfejs systemu.....	11
7. Testy i kontrola działania.....	11
III. PODSUMOWANIE I WNIOSKI.....	12
BIBLIOGRAFIA.....	13
SPIS TABEL.....	14
SPIS RYSUNKÓW.....	15