

# Overlord - Software Development Plan

Jakub Gołębiowski    Adam Kawa    Piotr Krewski    Tomasz Weksej

5 czerwca 2006

# Spis treści

0.1	Cel . . . . .	4
0.2	Zakres . . . . .	4
0.3	Definicje . . . . .	4
0.4	Załączniki . . . . .	4
0.5	Omówienie reszty dokumentu . . . . .	4
<b>1</b>	<b>Omówienie projektu</b>	<b>5</b>
1.1	Cel, zakres i 'objectives' projektu . . . . .	5
1.2	Założenia i zależności . . . . .	5
1.3	Produkty projektu . . . . .	5
<b>2</b>	<b>Organizacja projektu</b>	<b>5</b>
2.1	Struktura organizacyjna . . . . .	5
2.2	Role i odpowiedzialności . . . . .	6
<b>3</b>	<b>Zarządzanie projektem</b>	<b>7</b>
3.1	Oszacowania . . . . .	7
3.2	Plan projektu . . . . .	7
3.2.1	Plan faz i harmonogram projektu . . . . .	7
3.2.2	Diagram Gantta - faza projektowa . . . . .	9
3.2.3	Diagram Gantta - faza projektowa (ścieżka krytyczna) . . . . .	10
3.2.4	Diagram Gantta - faza projektowa (podział prac w zespole) . . . . .	11
3.2.5	Diagram Gantta - faza implementacyjna . . . . .	12
3.2.6	Diagram Gantta - faza implementacyjna (ścieżka krytyczna) . . . . .	12
3.2.7	Diagram Gantta - faza implementacyjna (podział prac w zespole) . . . . .	13
3.2.8	Wydania . . . . .	14
3.2.9	Zasoby . . . . .	14
3.2.10	Budżet . . . . .	14
3.3	Nadzór i kontrola projektu . . . . .	14
3.3.1	Plan zarządzania wymaganiami . . . . .	14
3.3.2	Plan zarządzania harmonogramem . . . . .	15
3.3.3	Plan kontroli jakości . . . . .	15
3.4	Plan zarządzania ryzykiem . . . . .	15
3.5	Plan zamknięcia projektu . . . . .	16
<b>4</b>	<b>Plany procesów technicznych</b>	<b>16</b>
4.1	Programowanie . . . . .	16
4.2	Metody, narzędzia i stosowane technologie . . . . .	16
4.3	Plan infrastruktury . . . . .	16
4.4	Plan zarządzania zmianami . . . . .	16
4.5	Plan oceny . . . . .	16

4.6 Plan rozwiązywania problemów . . . . .	16
<b>5 Historia zmian</b>	<b>16</b>

Dokument ten zawiera plan prac, niezbędnych do realizacji projektu OverLord.

## **0.1 Cel**

Celem tego dokumentu jest szczegółowe omówienie kwestii związanych z organizacją prac przy projekcie OverLord, .

## **0.2 Zakres**

Dokument ten obejmuje swoją treścią takie aspekty organizacji prac, jak podział zadań w obrębie zespołu projektowego, narzucenie harmonogramu dla tych zadań, a także analizę ryzyka związanego z projektem.

## **0.3 Definicje**

- BUC - Business Use Cases
- OVERLORD - nazwa tworzonego projektu
- SAD - Software Architecture Document.
- SDP - Software Development Plan (bieżący dokument)
- VISION - Wizja projektu OVERLORD

## **0.4 Załączniki**

- VISION, v.XXX
- BUC , v.XXX
- SAD , v.XXX

## **0.5 Omówienie reszty dokumentu**

Ten dokument zawiera następujące informacje:

- Omówienie projektu - dostarcza opis celów oraz zakresu projektu.
- Organizacja projektu - opisuje strukturę organizacyjną zespołu pracującego nad projektem.
- Zarządzanie projektem - określa przybliżony koszt i czas projektu, opisuje jak projekt będzie nadzorowany.

# 1 Omówienie projektu

## 1.1 Cel, zakres i 'objectives' projektu

Celem projektu jest przygotowanie w pełni funkcjonalnego systemu spełniającego wymagania przedstawione w dokumencie wizji. Projekt obejmuje dokumentację, implementację, testowanie oraz wdrożenie.

## 1.2 Założenia i zależności

Projekt jest realizowany przez czteroosobowy zespół studentów informatyki II roku Uniwersytetu Warszawskiego w ramach trwających jeden semestr zajęć akademickich. W trakcie trwania prac skład grupy realizującej projekt nie będzie podlegał żadnym zmianom. OVERLORD ma być autonomicznym projektem, nie będącym składową żadnego innego systemu informatycznego. Gracze nie potrzebują żadnych komercyjnych aplikacji by móc cieszyć się z pełni funkcjonalności gry, jedynym wymogiem jest przeglądarka www, poprzez którą następować będzie zarządzanie wirtualnym państwem.

## 1.3 Produkty projektu

- Wizja
- Biznesowe Przypadki Użycia
- Dokument Architektury Systemu
- Plan Projektu
- Aplikacja
- Zbiór Testów
- Podsumowanie Testów
- Adnotacje do Wydania

# 2 Organizacja projektu

## 2.1 Struktura organizacyjna

Projekt OVERLORD będzie realizowany przez zespół w składzie:

- Jakub Gołębiowski (jg219437@students.mimuw.edu.pl)
- Adam Kawa (ak221623@students.mimuw.edu.pl)

- Piotr Krewski (pk219456@students.mimuw.edu.pl)
- Tomasz Weksej (tw219722@students.mimuw.edu.pl)

## 2.2 Role i odpowiedzialności

Rola	Osoba	Zadania
Kierownik Zespołu	Tomasz Weksej	<ul style="list-style-type: none"> <li>• kontrola i podział pracy zespołu</li> <li>• kontakty zewnętrzne</li> </ul>
Projektant, specjalista ds. eksploatacji	Jakub Gołębiowski	<ul style="list-style-type: none"> <li>• definiuje cykl życia planu, oraz wszelkie koncepcje eksploatacyjne</li> </ul>
Projektant, specjalista ds. wymagań	Adam Kawa	<ul style="list-style-type: none"> <li>• definiuje cykl życia planu, oraz wymagania systemowe</li> </ul>
Projektant, specjalista ds. architektury	Piotr Krewski	<ul style="list-style-type: none"> <li>• definiuje wszelkie koncepcje dotyczące architektury oprogramowania i systemu</li> </ul>
Programista	Jakub Gołębiowski, Adam Kawa, Piotr Krewski, Tomasz Weksej	<ul style="list-style-type: none"> <li>• odpowiada za implementacje i działanie przydzielonej mu części aplikacji</li> </ul>
Tester	Jakub Gołębiowski, Adam Kawa, Piotr Krewski, Tomasz Weksej	<ul style="list-style-type: none"> <li>• planuje, przeprowadza i raportuje testy</li> <li>• wykrywa błędy krytyczne i niezgodności ze specyfikacją</li> </ul>

## 3 Zarządzanie projektem

### 3.1 Oszacowania

- planowany czas ukończenia prac nad dokumentacją - koniec maja 2006
- planowany czas ukończenia prac nad projektem - kwiecień 2007

### 3.2 Plan projektu

#### 3.2.1 Plan faz i harmonogram projektu

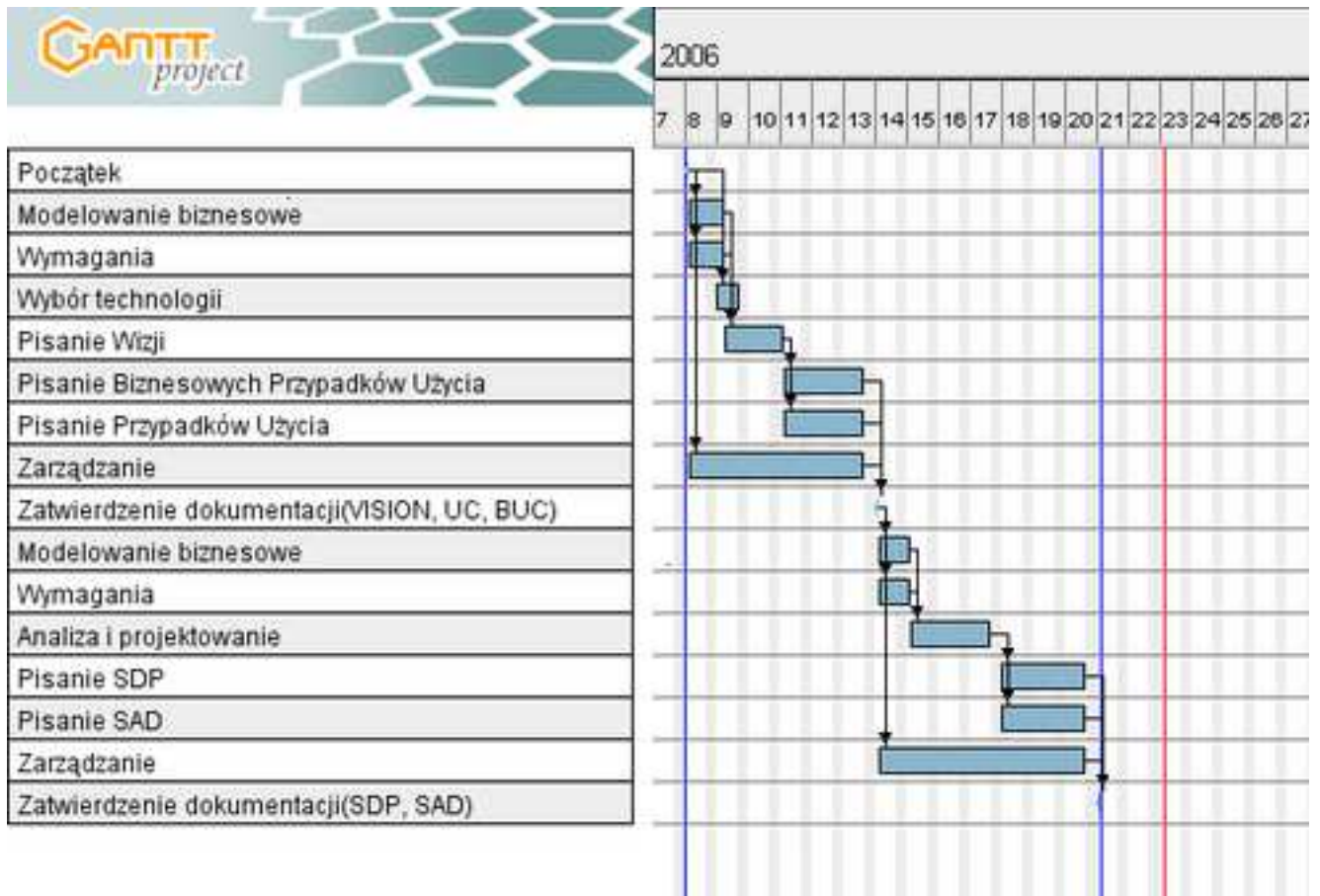
Nazwa	Rozpoczęcie	Zakończenie	Szczegóły
Sprecyzowanie wymagań biznesowych	20.02.06	28.02.06	
Wybór technologii	26.02.06	5.03.06	
Tworzenie dokumentacji - Vision, Use Cases, Business Use Cases	1.03.06	31.03.06	<ul style="list-style-type: none"><li>• podział prac nad dokumentacją</li><li>• sprawdzanie spójności i zgodności dokumentów</li></ul>
Tworzenie dokumentacji - Acceptance Plan, Software Architecture Document, Software Development Plan	1.04.06	20.05.06	<ul style="list-style-type: none"><li>• tworzenie dokumentów</li><li>• konsultacje z zespołem projektowym w sprawie merytorycznej zawartości dokumentów</li></ul>
Implementacja modułów	03.10.06	20.01.07	
Stworzenie wersji BETA	05.01.07	20.01.07	<ul style="list-style-type: none"><li>• wydanie pierwszej zawierającej kilka funkcjonalności, spójnej wersji systemu</li></ul>

Testowanie i poprawki systemu - ukończenie projektu	20.02.07	20.04.07	<ul style="list-style-type: none"><li>• przeprowadzanie na bieżąco testów przez testerów</li><li>• zlokalizowanie ewentualnych problemów i naniesienie korekt</li><li>• wydanie ostatecznej wersji systemu</li></ul>
---	----------	----------	--

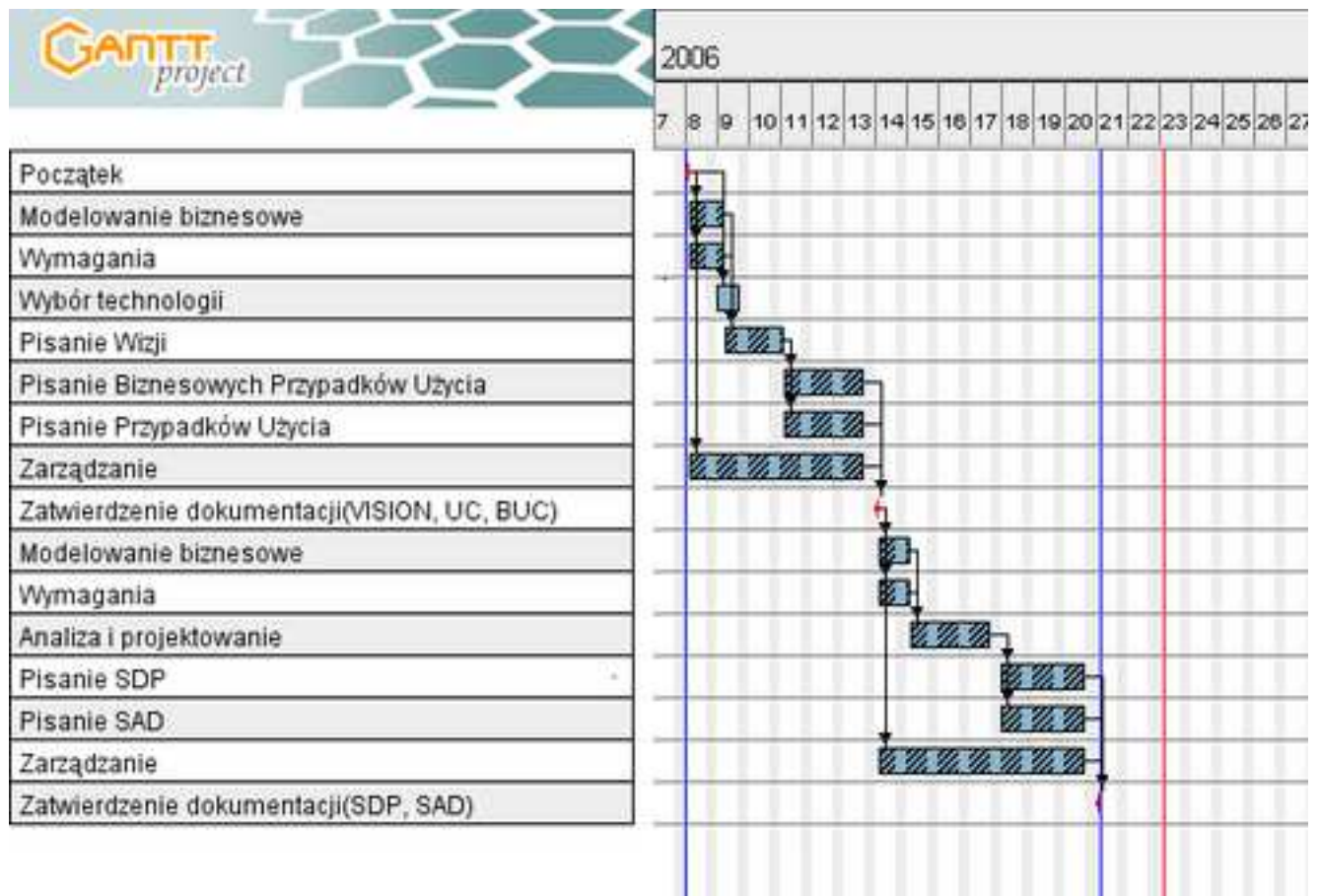
Kamienie Milowe :

- Początek
- Zatwierdzenie dokumentacji (VISION, UC, BUC)
- Zatwierdzenie dokumentacji (SDP, SAD)
- Wydanie wersji BETA
- Wydanie ostatecznej wersji

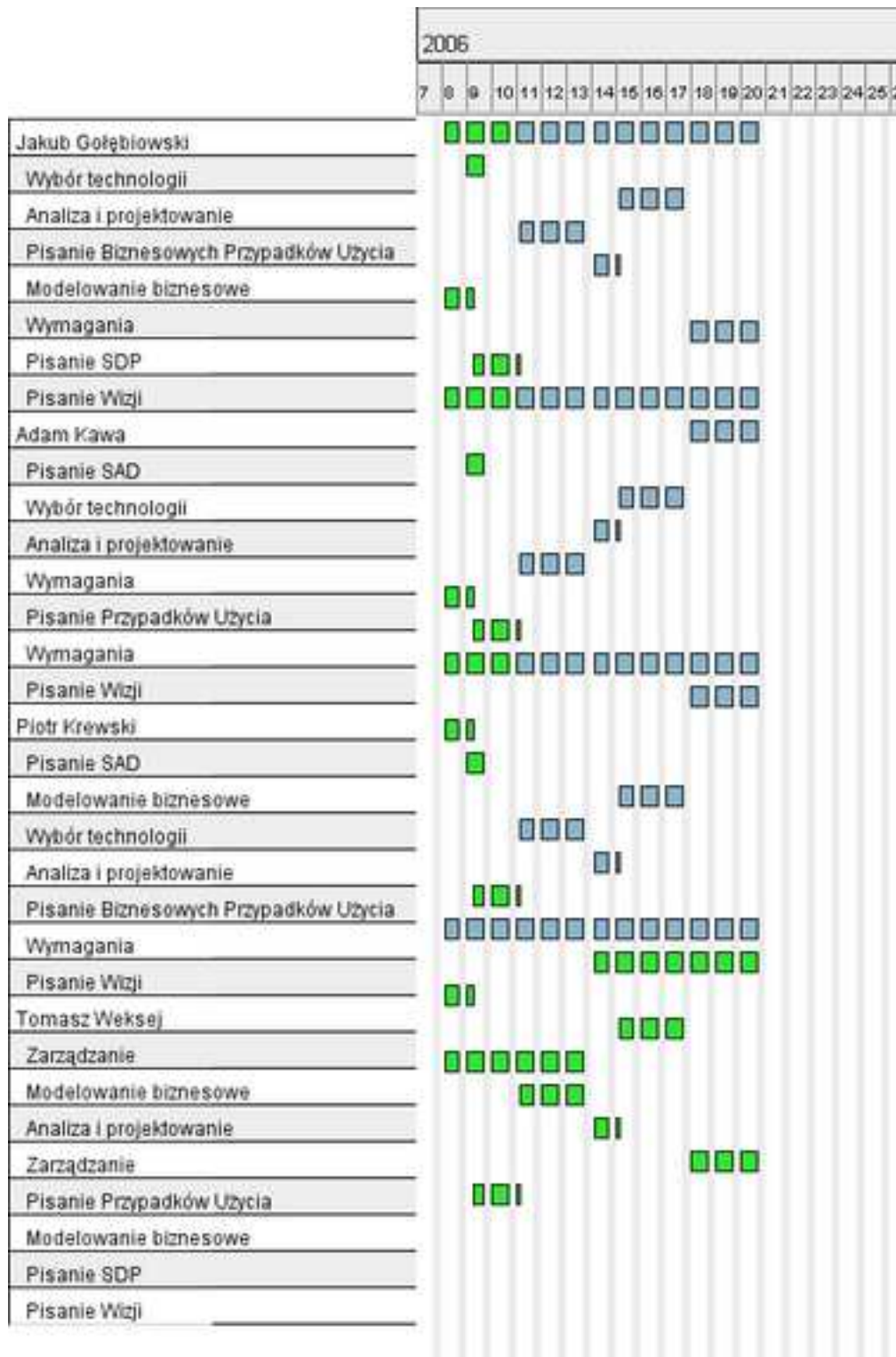
### 3.2.2 Diagram Gantta - faza projektowa



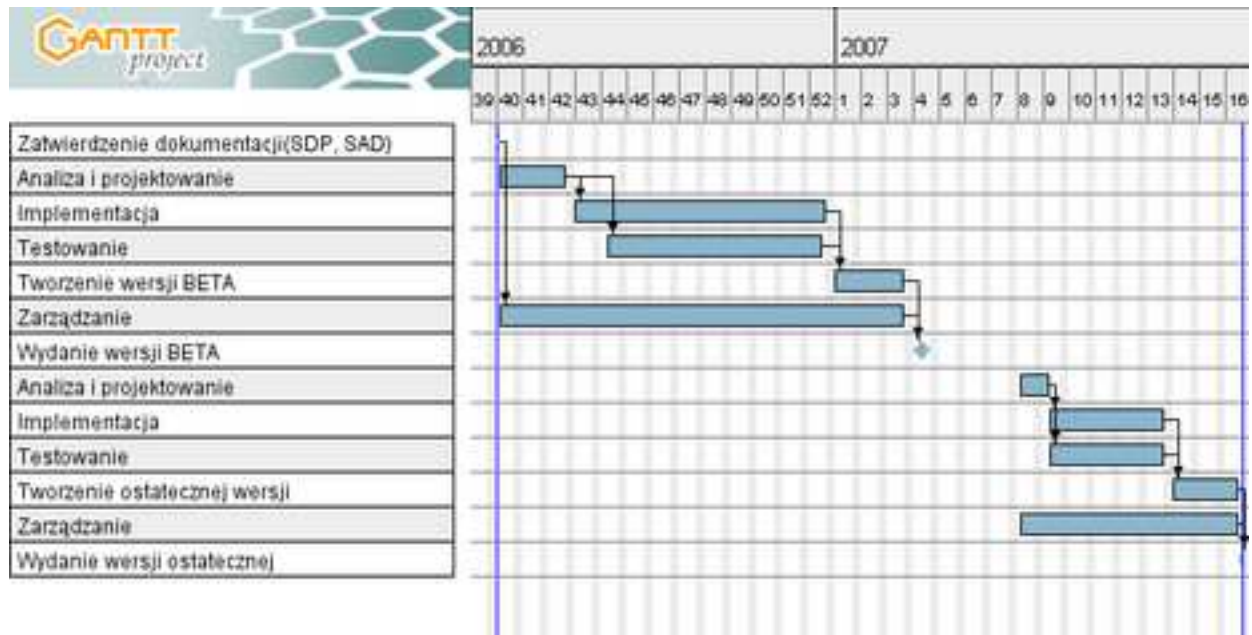
### 3.2.3 Diagram Gantt - faza projektowa (ścieżka krytyczna)



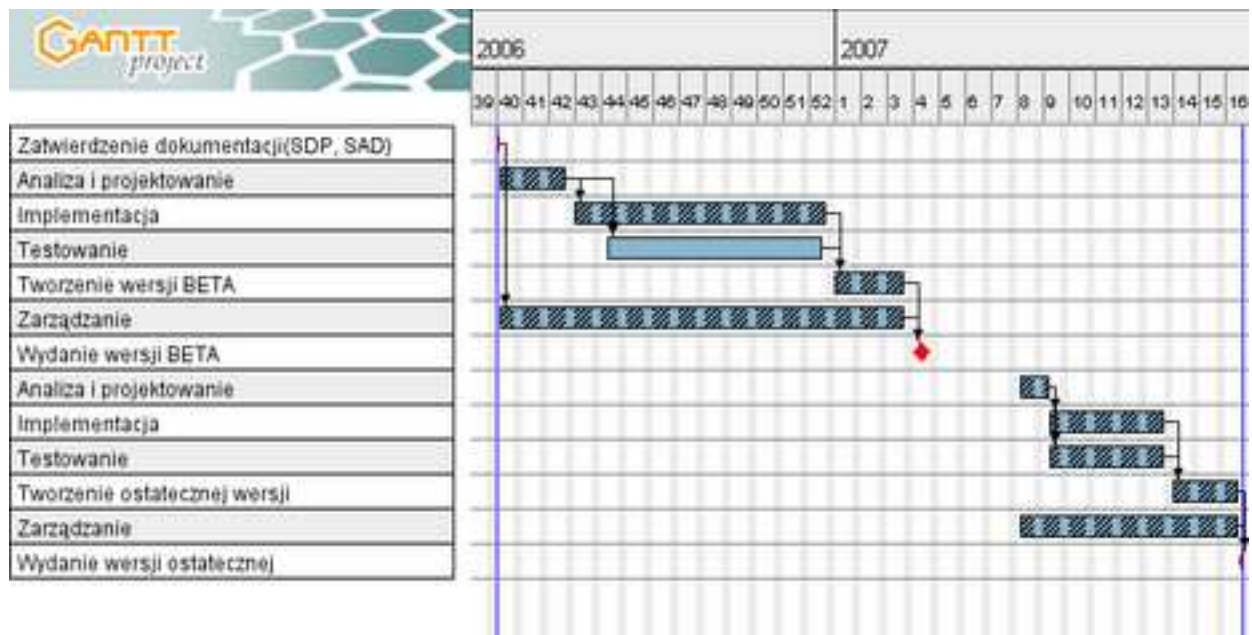
### 3.2.4 Diagram Gantta - faza projektowa (podział prac w zespole)



### 3.2.5 Diagram Gantta - faza implementacyjna



### 3.2.6 Diagram Gantta - faza implementacyjna (ścieżka krytyczna)





### 3.2.8 Wydania

Na przełomie grudnia 2006 i stycznia 2007 planujemy wydanie wersji beta, w której dostępne byłoby kilka najważniejszych funkcjonalności. W kwietniu pojawi się ostateczna wersja systemu.

### 3.2.9 Zasoby

**Plan zatrudnienia** Realizacją projektu zajmować się będzie czterech studentów informatyki UW. Dla powodzenia projektu będą oni musieli spełniać następujące wymagania:

- podstawowa znajomość języka Java
- nastawienie i umiejętność pracy w zespole
- pracowitość, rzetelność oraz umiejętność komunikacji
- osobowość - konflikty w zespole mogą bardzo poważnie utrudnić realizację projektu, stąd wymagamy aby każdy członek zespołu podporządkował się wspólnej wizji projektu
- umiejętność szybkiego rozwiązywania problemów - ze względu na wielkość projektu pojawiają się w nim wszelkiego rodzaju problemy, które będzie trzeba pokonać

**Plan zatrudniania pracowników** Zatrudnieni pracownicy tworzą zespół na zajęciach z Inżynierii Oprogramowania w semestrze letnim r. akad. 2005/2006 oraz na zajęciach Zespołowy Projekt Programistyczny w roku akademickim 2006/2007.

**Plan szkoleń** Członkowie zespołu będą musieli we własnym zakresie udoskonalać swoje umiejętności w programowaniu w języku Java oraz zapoznawać się ze standardami komunikacji sieciowej.

### 3.2.10 Budżet

Praca jest wykonywana nieodpłatnie na sprzęcie własnym, bądź udostępnionym nieodpłatnie przez Wydział Matematyki, Informatyki i Mechaniki Uniwersytetu Warszawskiego. Z tego powodu projekt nie wymaga żadnego wkładu finansowego.

## 3.3 Nadzór i kontrola projektu

### 3.3.1 Plan zarządzania wymaganiami

Wymagania mogą się zmienić jedynie w wyniku decyzji członków zespołu, którzy są jednocześnie wykonawcami i klientami.

### 3.3.2 Plan zarządzania harmonogramem

Ponieważ praca w ramach projektu jest wykonywana dobrowolnie oraz nie ma możliwości zmian w zespole, harmonogram będzie dostosowywany do wszystkich członków zespołu.

### 3.3.3 Plan kontroli jakości

Rodzaj	Częstotliwość	Cel
przeгляд spójności dokumentacji	raz na 3 tygodnie	wykrycie sprzeczności w dokumentacji
przeгляdy postępu	raz na 12 tygodni	potwierdzenie przebiegu prac zgodnie z harmonogramem, wyjaśnianie wątpliwości, wzajemna mobilizacja do pracy
testowanie	równolegle z implementacją, również po ukończeniu pierwszej wersji systemu	wykrycie błędów
walidacja	raz na 23 tygodnie	wykrycie rozbieżności między implementacją a założeniami i dokumentacją

### 3.4 Plan zarządzania ryzykiem

Nazwa	Skutki	Stopień ryzyka	Przeciwdziałanie
nadmiar obowiązków związanych z uczelnią	opóźnienia w pracy	średni	mądra organizacja czasu, przeгляdy postępu
problem z opanowaniem technologii	trudności z realizacją powierzonych zadań	niski	konsultacje w obrębie zespołu oraz zewnętrzne, wymiana zadań, stały kontakt
konflikty pomiędzy członkami zespołu	paraliż prac	niski	regularna integracja członków zespołu
słaby przepływ informacji	opóźnienia	niski	spotkania na zajęciach, używanie telefonów komórkowych, emaili, komunikatorów internetowych

### **3.5 Plan zamknięcia projektu**

Po zakończeniu projektu zostanie zwołana konferencja dla całego zespołu, na której omówione zostaną nabyte doświadczenia.

## **4 Plany procesów technicznych**

### **4.1 Programowanie**

### **4.2 Metody, narzędzia i stosowane technologie**

### **4.3 Plan infrastruktury**

### **4.4 Plan zarządzania zmianami**

Każda propozycja zmian jest dyskutowana na forum zespołu. Przedstawiane są skutki, jakie niesie za sobą dana zmiana i w przypadku jej zaaprobowania wybrany członek zespołu dokonuje stosownych poprawek w dokumentacji lub kodzie.

### **4.5 Plan oceny**

### **4.6 Plan rozwiązywania problemów**

Pojawiające się problemy będą dyskutowane na regularnych zebraniach zespołu i rozwiązywane w sposób demokratyczny po wyszczególnieniu zalet i wad możliwych rozwiązań.

## **5 Historia zmian**

\$Log: sdp.tex,v \$

Revision 1.6 2006/05/29 20:54:33 jg219437

Ukonczony punkt: Zarządzanie projektem.

Wersja do skontrolowania błędów.

Revision 1.5 2006/05/29 18:26:47 jg219437

Czesciowe opracowanie punktu: Zarządzanie projektem.

Revision 1.4 2006/05/28 19:36:02 jg219437

Przygotowany punkt: Organizacja Projektu

Revision 1.3 2006/05/25 18:10:16 jg219437

SDP - przygotowany punkt: Omowienie projektu.

Revision 1.2 2006/05/25 17:35:01 jg219437  
Przygotowany punkt: Wprowadzenie.

Revision 1.1 2006/05/25 10:01:29 jg219437  
Dodany szablon dokumentu sdp DO UZUPELNIENIA!!!