
Plan zarządzania projektem *Concerto*

Data 2013-01-06
Wersja 1.0
Autorzy TC, DG, ŁG, MM, MO, AO, MS

Spis treści

1	Wprowadzenie	2
1.1	Cel dokumentu	2
1.2	Zakres dokumentu	2
1.3	Dokumenty powiązane	2
2	Definicje	3
3	Zarys projektu	4
3.1	Kontekst projektu	4
3.2	Cel projektu	4
3.2.1	Dodatkowe cele techniczne	4
3.2.2	Dodatkowe cele nietechniczne	4
3.3	Założenia opłacalności	5
3.4	Szacowany budżet i wymagane zasoby	5
3.4.1	Zasoby techniczne	5
3.4.2	Zasoby czasowe	5
3.4.3	Zasoby ludzkie	5
4	Produkty projektu	6
4.1	Aplikacja mobilna	6
4.2	Serwer obsługi zapytań	6
4.3	API zarządzania bazą wydarzeń	6
4.4	Dokumentacja techniczna API	6
4.5	Współpraca produktów	6
5	Model procesu projektowego	8
5.1	Funkcje odbioru produktów	8
5.2	Synchronizacja wersji produktów	8
6	Organizacja projektu	9
6.1	Struktura organizacyjna	9
6.2	Plan zatrudnienia	9
6.3	Granice organizacyjne	10
6.4	Podział odpowiedzialności	10
7	Zarządzanie	14
7.1	Cele i priorytety zarządzania	14
7.2	Zarządzanie ryzykiem	14
7.3	Mechanizmy śledzenia i kontroli	14
7.4	Plan zatrudnienia	14

8	Proces techniczny	15
8.1	Metody, narzędzia i techniki	15
8.1.1	Serwer obsługi zapytań i API	15
8.1.2	Aplikacja mobilna	15
8.2	Dokumentacja oprogramowania	15
9	Harmonogram pracy	17
9.1	Podział projektu na etapy i zadania	17
9.1.1	Iteracja A – Przygotowanie	17
9.1.2	Iteracja B – Prototyp	17
9.1.3	Iteracja C – Wdrożenie	17

1 Wprowadzenie

1.1 Cel dokumentu

Dokument zawiera główne elementy planu zarządzania projektem *Concerto*. Jest głównym źródłem informacji o projekcie oraz planu pracy dla zespołu projektowego.

1.2 Zakres dokumentu

Dokument opisuje kontekst projektu, jego cele i oczekiwane produkty. Ustanawia organizację zespołu projektowego. Przedstawia kolejne etapy rozwoju projektu, harmonogram pracy i niezbędne zasoby.

1.3 Dokumenty powiązane

W ramach zarządzania projektem wykorzystywane będą dodatkowe dokumenty:

1. mapa pamięci projektu – obrazuje najważniejsze elementy projektu;
2. status projektu – dokument zawierający szczegółowy opis elementów projektu z mapy pamięci;
3. dokument zarządzania ryzykiem – opis zidentyfikowanych ryzyk specyficznych dla projektu *Concerto*, z propozycjami ich mitygacji;
4. dokument kontroli jakości – lista cech wymaganych do uzyskania produktów wysokiej jakości, wraz z metrykami i skalami.
5. specyfikacja funkcjonalna produktów – powstanie w czasie tworzenia projektu.

2 Definicje

Termin	Synonimy	Definicja
organizator współpracujący	współpracownik	Firma organizująca wydarzenia kulturalne, albo firma odpowiadająca za przeprowadzanie akcji reklamowych dla wydarzeń kulturalnych. Firmy, do których skierowana jest biznesowa oferta projektu <i>Concerto</i> .
użytkownik	konsument	Osoba będąca adresatem akcji reklamowych. Docelowy użytkownik aplikacji mobilnej.
zespół projektowy	zespół <i>Concerto</i>	Zespół pracowników firmy tworzącej i obsługującej produkty projektu <i>Concerto</i> .
serwer obsługi zapytań	serwer	Usługa internetowa obsługująca zapytania aplikacji mobilnej. Dostępna z maszyny fizycznej lub wirtualnej, zależnie od etapu prac nad projektem.
aplikacja mobilna		Aplikacja na urządzenia mobilne przeznaczona dla adresatów akcji reklamowych wydarzeń kulturalnych.
API	interfejs API; interfejs zarządzania wydarzeniami	Usługa internetowa pozwalająca na zarządzanie bazą wydarzeń kulturalnych serwera. Udostępniania wyłącznie organizatorom, z którymi nawiązano współpracę.

3 Zarys projektu

3.1 Kontekst projektu

Kampus Politechniki Warszawskiej jest popularnym miejscem działań agencji marketingowych promujących wydarzenia kulturowe, w szczególności koncerty. Wybór miejsca jest podyktowany kilkoma czynnikami:

- studenci są główną grupą docelową tego typu wydarzeń,
- kampus zapewnia duży przepływ osób z grupy docelowej,
- studenci są aktywnymi konsumentami kultury — łatwo jest ich zaangażować w akcjach promocyjnych.

W efekcie, nie potrzeba wiele nakładów by wypromować koncert na kampusie Politechniki Warszawskiej. Sprawdzoną i wystarczającą metodą promocji jest wywieszenie plakatów promujących wydarzenie w widocznych miejscach, na przykład przed wejściami do Gmachu Głównego. Tak umieszczony plakat wystarczy, by przekonać pojedynczych studentów. Oni natomiast zaproszą osobiście swoich znajomych. Jedynym problemem tego scenariusza jest zawodna pamięć studentów, zwłaszcza gdy zobaczyli plakat biegnąc na wykład, by uniknąć spóźnienia.

3.2 Cel projektu

Celem projektu *Concerto* jest **zwiększenie efektywności plakatowych akcji reklamowych** wydarzeń kulturalnych. W ramach projektu powstanie system udostępniający oprogramowanie jako usługę. Usługa będzie oferowana organizatorom wydarzeń kulturalnych.

3.2.1 Dodatkowe cele techniczne

Usługa *Concerto* wymaga stworzenia systemu niezależnych, współpracujących komponentów:

1. aplikacji mobilnej dla adresatów akcji reklamowych,
2. serwera obsługi zapytań od aplikacji mobilnych,
3. API udostępniane organizatorom do zarządzania informacjami o reklamowanych wydarzeniach kulturalnych.

Dokładny opis komponentów znajduje się na stronie 6.

3.2.2 Dodatkowe cele nietechniczne

Utrzymanie infrastruktury i zespołu projektu wymaga:

1. nawiązania współpracy z organizatorami, na warunkach przynoszących zysk, pomimo stosunkowo wysokich kosztów utrzymania infrastruktury systemu;
2. uzyskania stałej bazy użytkowników aplikacji mobilnej, potrzebnej przy negocjowaniu warunków współpracy z użytkownikami.

3.3 Założenia opłacalności

Źródłem przychodów jest prowizja od sprzedaży biletów, które zostały dokonane z polecenia aplikacji mobilnej *Concerto*. Wysokość prowizji jest ustalana podczas nawiązywania współpracy z organizatorami wydarzeń. Warunki współpracy zależą od:

- fazy wdrożenia projektu. Wczesna współpraca zakłada lepsze warunki dla organizatorów. Zespołowi projektowemu daje to możliwość uniknięcia konieczności zaciągnięcia kredytu.
- aktualnego rozmiaru bazy użytkowników aplikacji mobilnej.

Wstępnie rozwój usługi zakłada nawiązywanie współpracy z możliwie największą grupą organizatorów wydarzeń. W przypadku napotkania problemów z uzyskaniem zadowalających przychodów możliwa jest zmiana polityki rozwoju. Alternatywną ścieżką rozwoju jest oferowanie współpracy tylko najważniejszym organizatorom wydarzeń z warunkiem ekskluzywności.

3.4 Szacowany budżet i wymagane zasoby

3.4.1 Zasoby techniczne

Podczas prac planowany jest jednorazowy zakup sprzętu serwerowego i reprezentatywnego zestawu urządzeń mobilnych potrzebnych w procesie testowania. Budżet na zasoby techniczne na potrzeby prac nad projektem wynosi 9 000 PLN.

3.4.2 Zasoby czasowe

Zespół ma jeden rok na realizację projektu.

3.4.3 Zasoby ludzkie

Zespół projektowy składa się z:

- zespołów programistycznych – 5 osób;
- manager projektu i produktu – 1 osoba;
- analityk – 1 osoba.

Wynagrodzenie zespołu, na czas tworzenia projektu, szacowane jest na 140 000 PLN.

4 Produkty projektu

System *Concerto* składa się trzech komponentów:

1. aplikacja mobilna dla adresatów akcji reklamowych,
2. serwer obsługi zapytań od aplikacji mobilnych,
3. API udostępniane organizatorom do zarządzania informacjami o reklamowanych wydarzeniach kulturalnych.

Schemat współpracy komponentów jest przedstawiony na stronie 7.

4.1 Aplikacja mobilna

Głównym celem aplikacji mobilnej jest dostarczenie użytkownikowi informacji o wydarzeniu oraz zaproponowanie kupna biletu w sklepie internetowym organizatora.

Po uruchomieniu aplikacji użytkownik robi zdjęcie plakatu, który go zainteresował. Aplikacja wysyła zdjęcie do serwera obsługi zapytań i oczekuje na odpowiedź. Zwracane są podstawowe informacje, takie jak czas i miejsce wydarzenia, cena biletu oraz propozycja zakupu biletu bezpośrednio w sklepie internetowym organizatora.

Następnie umożliwia zaproszenie znajomych na wydarzenie korzystając z popularnych sieci społecznościowych oraz standardowych środków komunikacji, takich jak email czy SMS.

4.2 Serwer obsługi zapytań

Serwer obsługi zapytań jest dostępny przez połączenie internetowe. Serwer posiada bazę danych z informacjami o wydarzeniach, w szczególności obrazy plakatów wykorzystywanych w akcjach reklamowych. Po otrzymaniu zapytania, serwer porównuje zdjęcie z plakatami z bazy i zwraca informacje przypisane do plakatu.

4.3 API zarządzania bazą wydarzeń

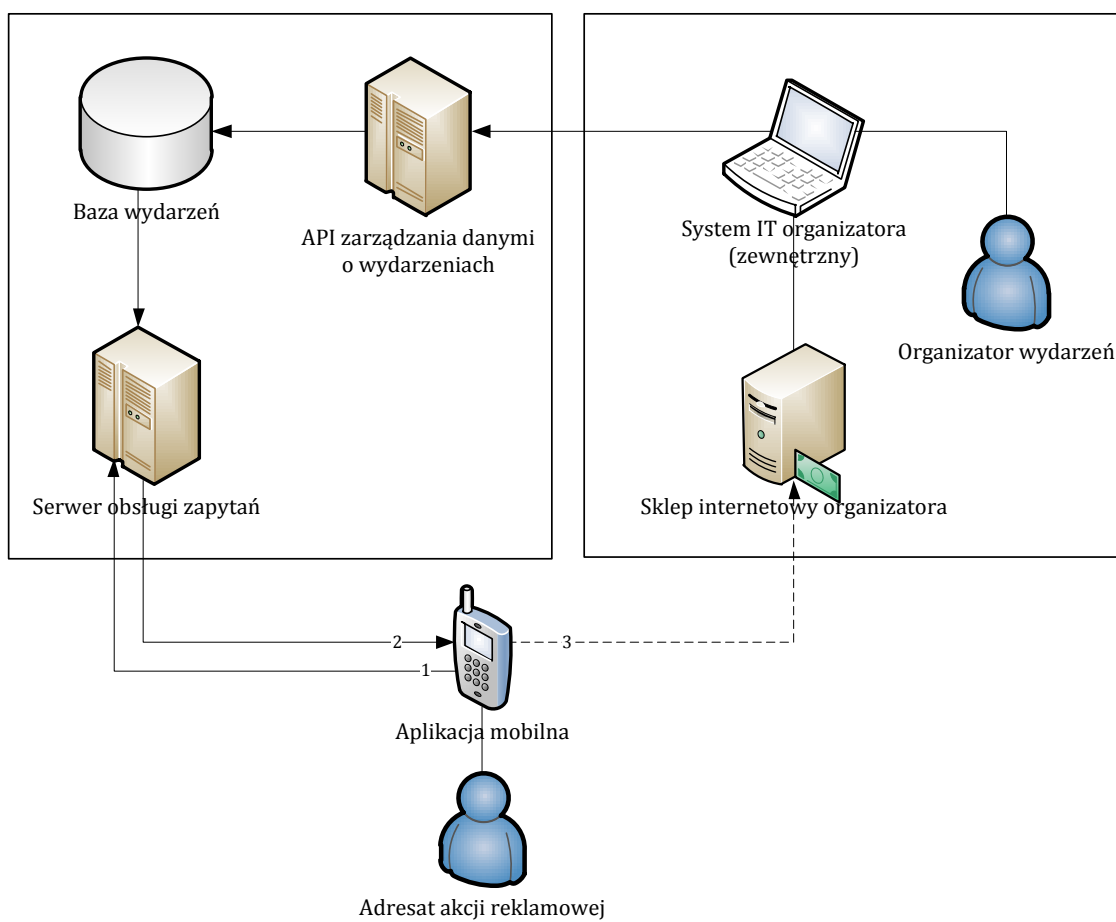
System ściśle współpracujący z serwerem obsługi zapytań. Pozwala organizatorom wydarzeń na zarządzanie plakatami i informacjami o wydarzeniach do bazy danych serwera obsługi zapytań.

4.4 Dokumentacja techniczna API

Dokumentacja przeznaczona dla współpracujących organizatorów. Zawiera opis interfejsu programistycznego, pozwalającego na zarządzanie danymi wydarzeń kulturalnych.

4.5 Współpraca produktów

Współpraca produktów jest przedstawiona na schemacie 1 na stronie 7.



Rysunek 1: Schemat współpracy wszystkich produktów technicznych projektu.

5 Model procesu projektowego

5.1 Funkcje odbioru produktów

Projekt jest tworzony i odbierany przez ten sam zespół projektowy. Ewentualne nadanie współpracującym organizatorom roli mającej wpływ na produkty lub cele projektu jest rozpoznany ryzykiem. Reakcją jest rozpatrzenie możliwości i ewentualne wdrożenie pivotu projektu, zależnie od warunków nawiązywanej współpracy.

Projekt realizowany jest w trzech głównych, następujących po sobie iteracjach. Zakończenie każdej kolejnej iteracji rozszerza funkcjonalność produktów. Planowane iteracje to:

- **Iteracja A** – Przygotowanie
- **Iteracja B** – Prototyp
- **Iteracja C** – Wdrożenie

Zadania i produkty iteracji *Przygotowanie* mają na celu stworzenie szczegółowej dokumentacji wspomagającej i organizującej prace nad projektem. Po zakończeniu tej iteracji rozpoczęte zostają równoległe prace nad produktami projektu. Procesy testowania oraz integracji zachodzą możliwie ciągle.

Planowane są dwa formalne przyjęcia produktów, po zakończeniu iteracji B i C. Akceptacja jest zadaniem zespołu projektowego. Ocena jest oparta o:

1. Wytyczne dokumentu kontroli jakości w postaci zestawu testów akceptacyjnych;
2. Wytyczne specyfikacji funkcjonalnej w postaci zestawu testów funkcjonalnych.

Opisana metoda oceny produktów jest stałym elementem procesu testowania, który zachodzi ciągle przez cały czas trwania prac. W efekcie zakończenie ostatniego zadania z danej iteracji, automatycznie kończy prace nad iteracją i zostaje ona przyjęta.

Do zakończenia iteracji *Prototyp* oczekiwana jest akceptacja krytycznych części funkcjonalności systemu. W przypadku nie uzyskania oceny pozytywnej przewidziane jest wydłużenie czasu trwania tej iteracji. Po zakończeniu iteracji *Wdrożenie* dokonywana jest ocena pełnej funkcjonalności systemu. W przypadku spełnienia wytycznych projekt jest uznawany za zakończony. W przeciwnym przypadku planowane są zadania uzupełniające oraz rozszerzenie budżetu i harmonogramu pracy.

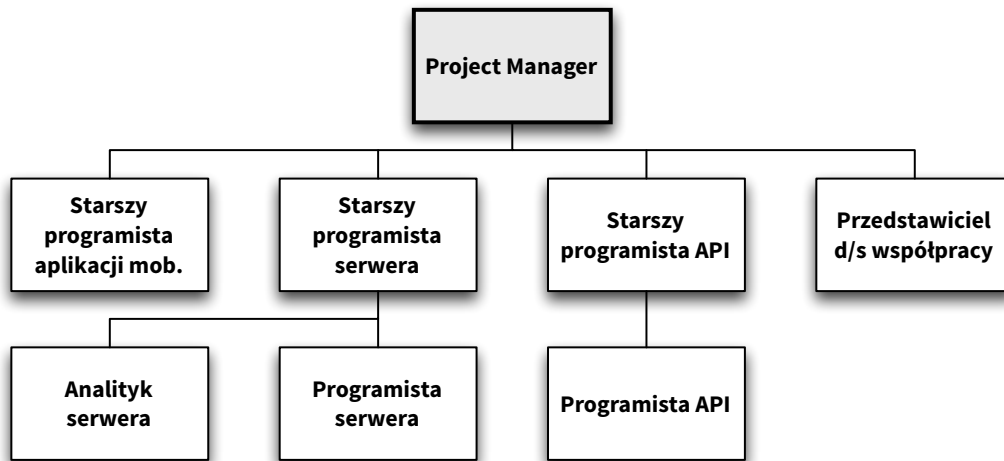
5.2 Synchronizacja wersji produktów

Synchronizacja wersji produktów jest prowadzona przez system kontroli wersji Git. Każdy produkt jest przechowywany w niezależnym repozytorium. Kiedy wszystkie produkty dojdą do milestone'ów synchronizujących określonych w harmonogramie pracy, ich milestone'owe rewizje zostają dodane do meta-repozytorium całego projektu. W ten sposób prace nad produktami mogą przebiegać niezależnie. Jednocześnie w pełni funkcjonalny produkt jest utrzymywany w meta-repozytorium głównym.

6 Organizacja projektu

6.1 Struktura organizacyjna

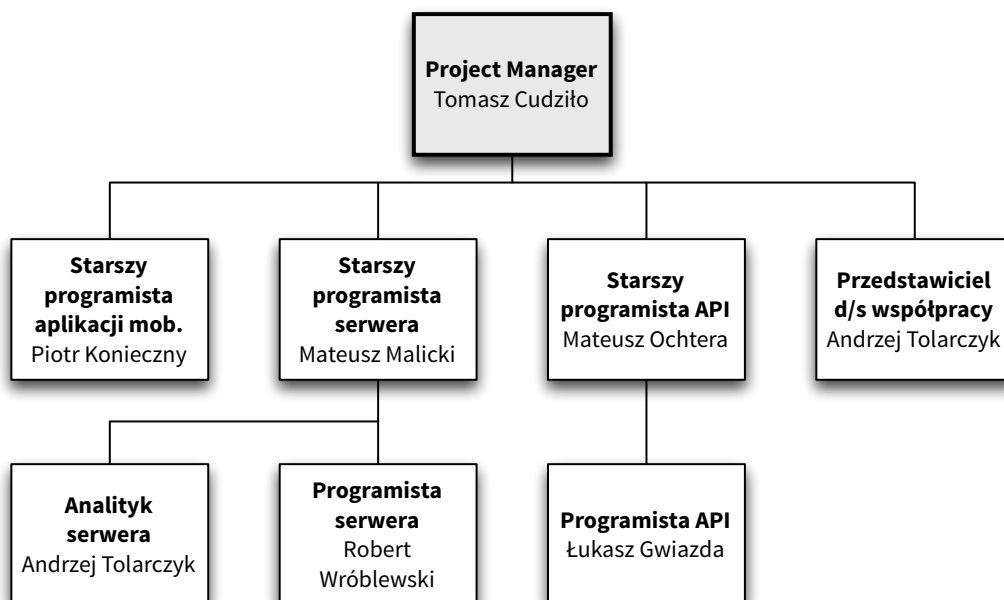
Struktura zespołu projektowego jest przedstawiona na rys. 2 na stronie 9.



Rysunek 2: Schemat struktury organizacyjnej zespołu projektowego.

6.2 Plan zatrudnienia

Planowane jest rozdzielenie ról według schematu z rys. 3. na stronie 9.



Rysunek 3: Schemat przydziału ról w zespole projektowym.

6.3 Granice organizacyjne

Zespół projektowy *Concerto* jest niezależną jednostką. Nawiązywanie współpracy wymaga kontaktu z przedstawicielem biznesowym organizatora wydarzeń kulturalnych. Po nawiązaniu współpracy, w wyjątkowych sytuacjach, komunikacja odbywa się z działem IT oraz osobą odpowiedzialną za przeprowadzane akcje reklamowe – przedstawicieli wyznaczonych ze strony organizatora wydarzeń. Produkt projektu – API stanowi podstawowy kanał komunikacyjny, dla standardowych zdarzeń.

6.4 Podział odpowiedzialności

Macierze odpowiedzialności za kolejne grupy zadań przewidzianych w pracach nad projektem są załączone zaczynając od strony 11.

Zadania organizacyjne, zarządzania projektem i reprezentowania zespołu projektowego, jeżeli nie uwzględnione w macierzach odpowiedzialności, są odpowiedzialnością Project Managera.

SPP	Zadanie	Tomasz Cudziło	Łukasz Gwiazda	Piotr Konieczny	Mateusz Malicki	Mateusz Ochtera	Andrzej Tolarczyk	Robert Wróblewski
A	Przygotowanie	A						
A.1	Koncept projektu	A						
A.2	Organizacja zespołu	A						
A.8	Koordynacja pracy	A						
A.4	Server obsługi zapytań				A		R	R
A.5	API		R			A		
A.6	Aplikacja mobilna			A				
A.7	Testowanie	A		R	R	R		
A.9	Administracja infrastruktury	A		R				
A.10	Nawiązywanie współpracy						A	

Legenda:

A = Accountable, Kierownik podzespołu

R = Responsible, Członek zespołu

C = Consultant, Wtyka kierownika

I = Informed, Śledzą wyniki prac

Rysunek 4: Macierz odpowiedzialności za zadania w iteracji *Przygotowanie*.

SPP	Zadanie	Tomasz Cudziło	Łukasz Gwiazda	Piotr Konieczny	Mateusz Malicki	Mateusz Ochtera	Andrzej Tolarczyk	Robert Wróblewski
B	Prototyp					A		
B.1	Server obsługi zapytań				A		C	R
B.2	Aplikacja mobilna			A				
B.3	API		R			A		
B.4	Testowanie	A						
B.5	Metryki warunków satysfakcji		R	R	R	A		R
B.6	Administracja infrastruktury	A						
B.7	Nawiązywanie współpracy						A	

Legenda:

A = Accountable, Kierownik podzespołu
R = Responsible, Członek zespołu
C = Consultant, Wtyka kierownika
I = Informed, Śledzą wyniki prac

Rysunek 4: Macierz odpowiedzialności za zadania iteracji *Prototyp*.

SPP	Zadanie	Tomasz Cudziło	Łukasz Gwiazda	Piotr Konieczny	Mateusz Malicki	Mateusz Ochtera	Andrzej Tolarczyk	Robert Wróblewski
C	Wdrożenie				A			
C.1	Server obsługi zapytań				A			R
C.2	Aplikacja mobilna			A				
C.3	API		R			A		
C.4	Testowanie	A			R	R		
C.6	Administracja infrastruktury	A			R	R		
C.7	Nawiązywanie współpracy						A	

Legenda:

- A = Accountable, Kierownik podzespołu
- R = Responsible, Członek zespołu
- C = Consultant, Wtyka kierownika
- I = Informed, Śledzą wyniki prac

Rysunek 4: Macierz odpowiedzialności za zadania iteracji *Wdrożenie*.

7 Zarządzanie

7.1 Cele i priorytety zarządzania

Głównymi czynnikami motywującymi akcje zarządzania projektem są:

- nie przekroczenie budżetu projektu,
- uzyskanie produktów wysokiej jakości.

Dotrzymanie terminów nie jest głównym priorytetem w planie zarządzania. Jest przewidziana swoboda z czasem przeznaczonym na pracę. Zespół wyznaczył czas tworzenia projektu na 1 rok, przy czym harmonogram prac zakłada ukończenie iteracji *Wdrożenie* po około 5 miesiącach od rozpoczęcia projektu.

7.2 Zarządzanie ryzykiem

Lista zidentyfikowanych ryzyk zawarta jest w załączonym *Dokumencie zarządzania ryzykiem*.

7.3 Mechanizmy śledzenia i kontroli

Zadania są przydzielane zgodnie z harmonogramem prac i podziałem odpowiedzialności. Kontrola nad postępami prac jest zapewniana przez obowiązkowe recenzje kodu każdej zaimplementowanej części funkcjonalności produktów. Recenzja ma miejsce przed włączeniem odpowiedzialnego kodu do repozytorium głównego. Zarządzanie kodem źródłowym w systemie kontroli wersji odbywa się zgodnie z metodyką *git-flow*. Gdy recenzja jest pozytywna, zadanie jest uznawane za zakończone i kolejne zadania dodawane są do aktualnej puli zadań oczekujących na implementację.

7.4 Plan zatrudnienia

Projekt jest tworzony przez siedmioosobowy zespół programistyczny:

- Tomasz Cudziło
- Łukasz Gwiazda
- Piotr Konieczny
- Mateusz Malicki
- Mateusz Ochtera
- Andrzej Tolarczyk
- Robert Wróblewski

Role przedzielone zgodnie z planem zatrudnienia z rys. 3 ze strony 9.

W trakcie tworzenia projektu planowane jest zatrudnienie zespołu testerów aplikacji mobilnej. Po zakończeniu iteracji *Wdrożenie* możliwe jest ograniczenie zespołu do osób niezbędnych do utrzymania pracy systemu. Jednocześnie możliwe jest przyjęcie dodatkowej osoby koordynującej nawiązywanie i negocjację współpracy z organizatorami wydarzeń. Pozwoli to na zwolnienie Project Managera z części obowiązków.

8 Proces techniczny

8.1 Metody, narzędzia i techniki

W ramach projektu zespół będzie stosował metodykę zwinną *Kanban*. Narzędzia wspomagające metodykę są częścią pakietu usług firmy *Atlassian*.

1. Repozytoria kodu źródłowego dla systemów kontroli wersji oraz wiki z dokumentami powiązanymi przetrzymywane są na platformie *Bitbucket*.
2. Lista zadań, śledzenie postępów oraz ewentualne dyskusje znajdują się na platformie *GreenHopper*.
3. Recenzje kodu odbywają się na poziomie głównego repozytorium. Włączenia kodu, prośby i realizacja jego recenzji odbywa się na *Bitbucket*.
4. Automatyczne testowanie i zarządzanie aktualną wersją oprogramowania serwerów produkcyjnych zarządzane są z platformy *Bamboo*.

Usługi są zintegrowane i dostępne wszystkim członkom zespołu projektowego. Komunikacja odbywa się w ramach wymienionych narzędzi, pomiędzy wszystkimi poziomami struktury organizacyjnej. Pozwala na to mały rozmiar zespołu projektowego oraz bezpośrednie zaangażowanie w prace techniczne nad produktami przez cały zespół.

8.1.1 Serwer obsługi zapytań i API

Do utworzenia serwera i API wykorzystany będzie stos technologiczny składający się z:

- Nginx, Unicorn – serwery, obsługa i balansowanie zapytań HTTP,
- Ruby on Rails – framework dla aplikacji serwerowej,
- PostgreSQL – baza danych,
- RSpec, Rake, RMagick – biblioteki języka Ruby oferujące potrzebną funkcjonalność.

8.1.2 Aplikacja mobilna

Do stworzenia aplikacji mobilnej na system operacyjny Windows Phone 8 zostanie wykorzystany standardowy zestaw bibliotek z pakietu .NET 4.5.

8.2 Dokumentacja oprogramowania

W ramach tworzenia projektu zostaną utworzone i będą wykorzystywane dokumenty:

- Szablony dokumentów – wymuszają wspólny styl na dokumentach projektu oraz zapewniają obecność wymaganych elementów.
- Dokument stylu kodu – do użytku wewnętrznego.
- Specyfikacja funkcjonalna produktów projektu – Powstaje podczas iteracji *Przygotowanie*. Jest podstawą do tworzenia specyfikacji technicznej, oraz dokładnych warunków testów akceptacyjnych i metryk z dokumentu kontroli jakości.
- Dokumentacja techniczna API – jeden z produktów iteracji *Wdrożenie*. Niezależny dokument, przeznaczony do użytku zewnętrznego przez współpracowników.
- Dokumentacja techniczna produktów – tworzona jest ciągle podczas trwania prac nad kodem. Opisuje interfejsy klas w kodzie źródłowym. Przeznaczona do użytku wewnętrznego przez zespół projektowy.

- Specyfikacja techniczna produktów – tworzona jest ciągle w ramach procesu testowania. Jest w formie wykonywalnego kodu, wchodzi w skład zestawu automatycznych testów funkcjonalnych.

9 Harmonogram pracy

9.1 Podział projektu na etapy i zadania

Prace projektowe odbywają się w trzech iteracjach.

9.1.1 Iteracja A – Przygotowanie

Ogólnym celem iteracji A jest przygotowanie zespołu do prac nad produktami projektu. Iteracja zakłada:

- ustalenie wspólnej wizji projektu,
- uzyskanie specjalistycznej wiedzy przez zespół,
- stworzenie podstawowej dokumentacji,
- utworzenie środowiska pracy.

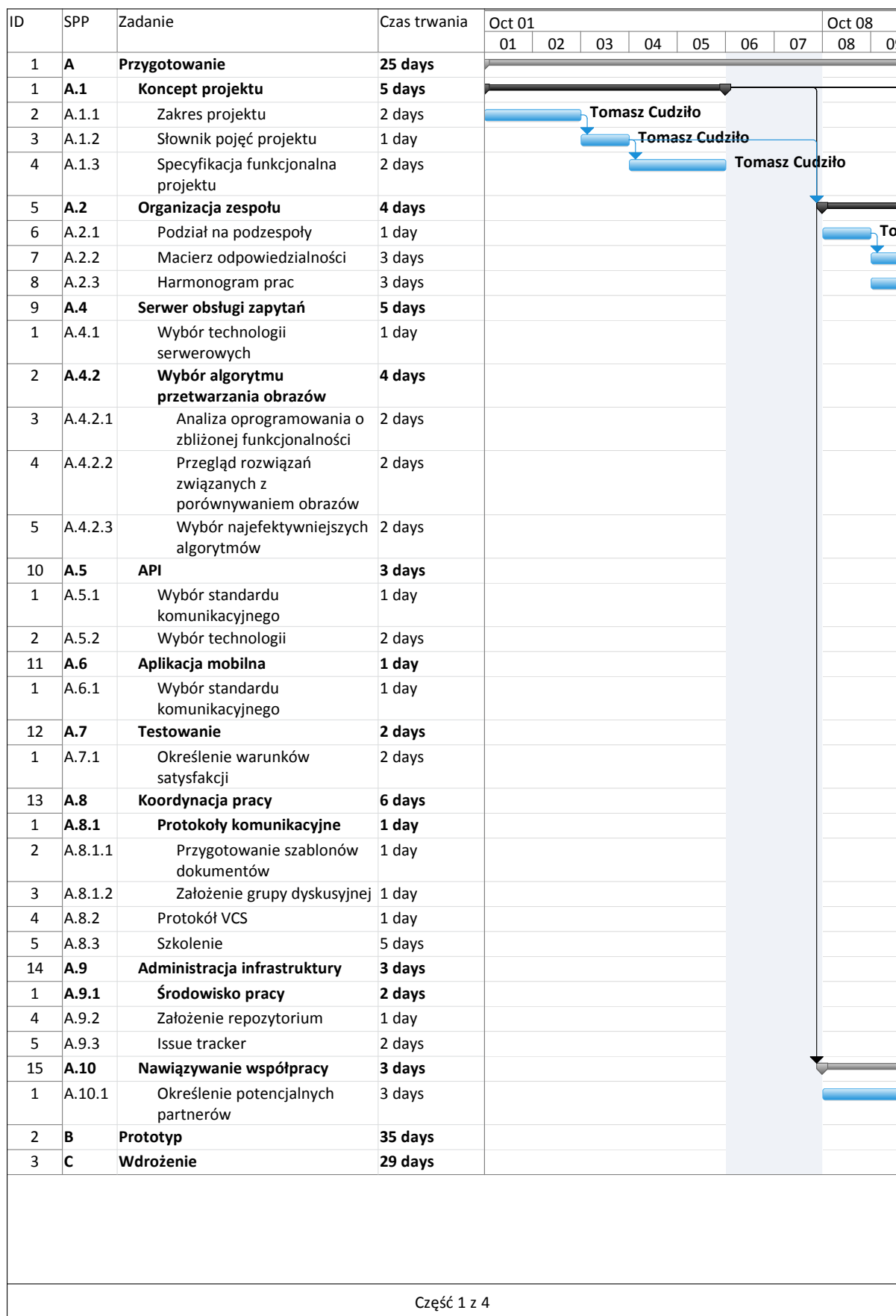
Planowane rozpoczęcie iteracji A jest na 1. października 2012, z końcem prac 3. listopada 2012. Budżet iteracji wynosi 40 000 PLN. Harmonogram wykonania zadań wchodzących w skład iteracji *Przygotowanie* zaczyna się na stronie 18.

9.1.2 Iteracja B – Prototyp

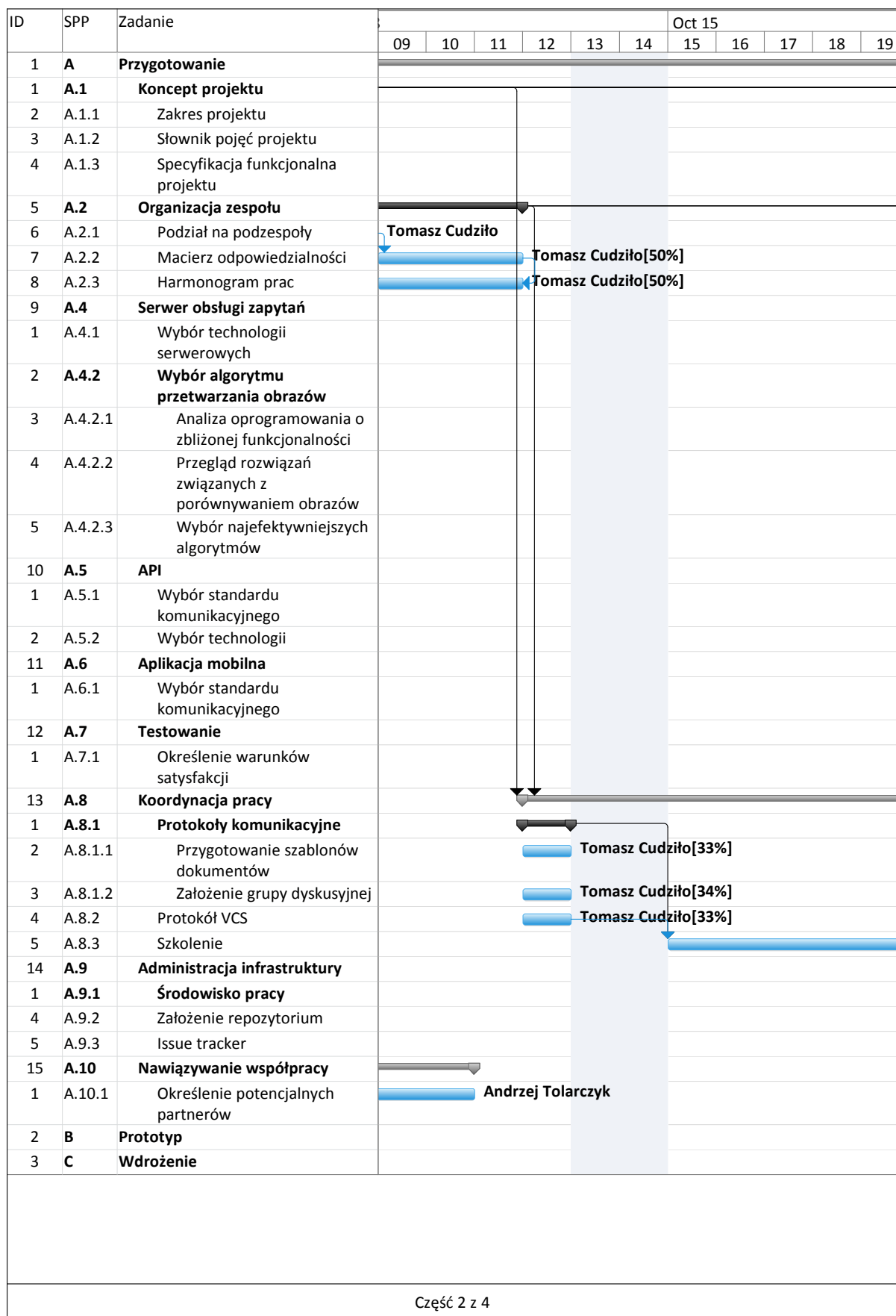
Zadania iteracji B mają na celu stworzenie produktów oferujących jedynie funkcjonalność krytyczną dla projektu. Iteracja B zaczyna się 5. listopada 2012 i kończy 21. grudnia 2012 r. Zakładany budżet to 53 000 PLN. Harmonogram prac nad zadaniami iteracji *Prototyp* zaczyna się na stronie 22.

9.1.3 Iteracja C – Wdrożenie

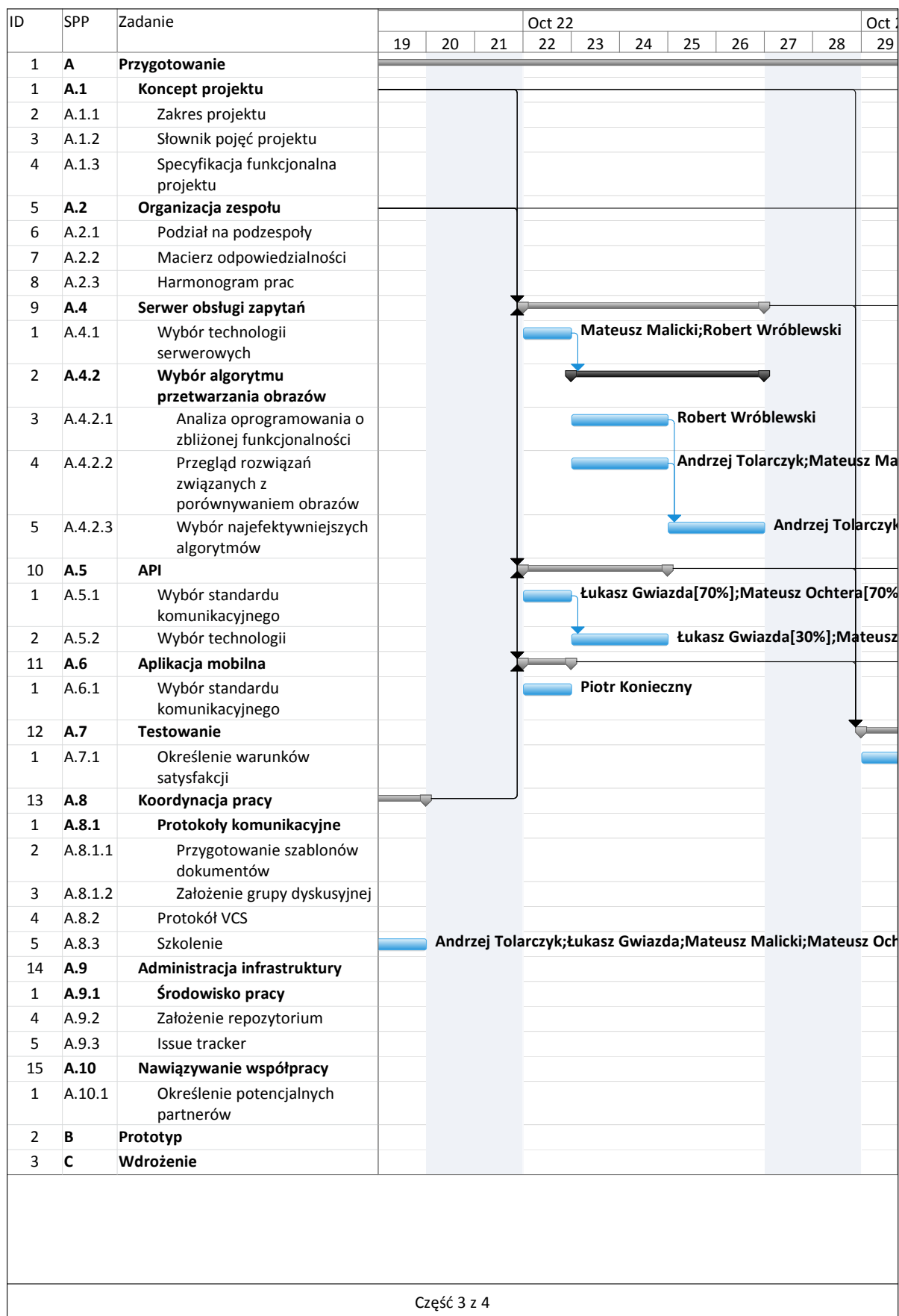
Prace iteracji C zaczynają się 3. stycznia 2013, a kończą 12. lutego 2013. Budżet iteracji to 57 000 PLN. Harmonogram przydziału zadań iteracji *Wdrożenie* zaczyna się na stronie 27.



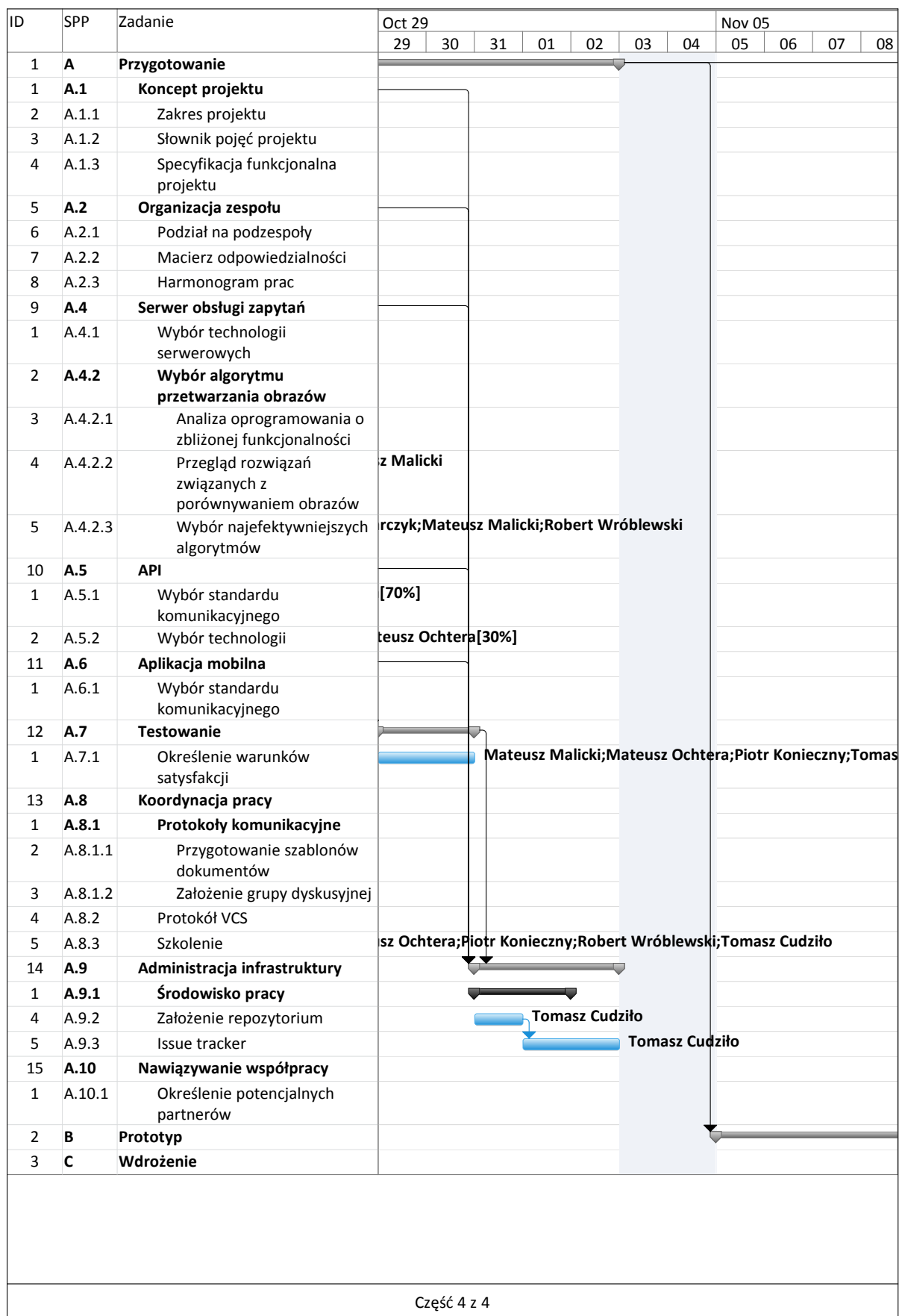
Rysunek 5: Harmonogram pracy iteracji *Przygotowanie projektu Concerto*.



Rysunek 5: Harmonogram pracy iteracji *Przygotowanie projektu Concerto*. (kontynuacja)

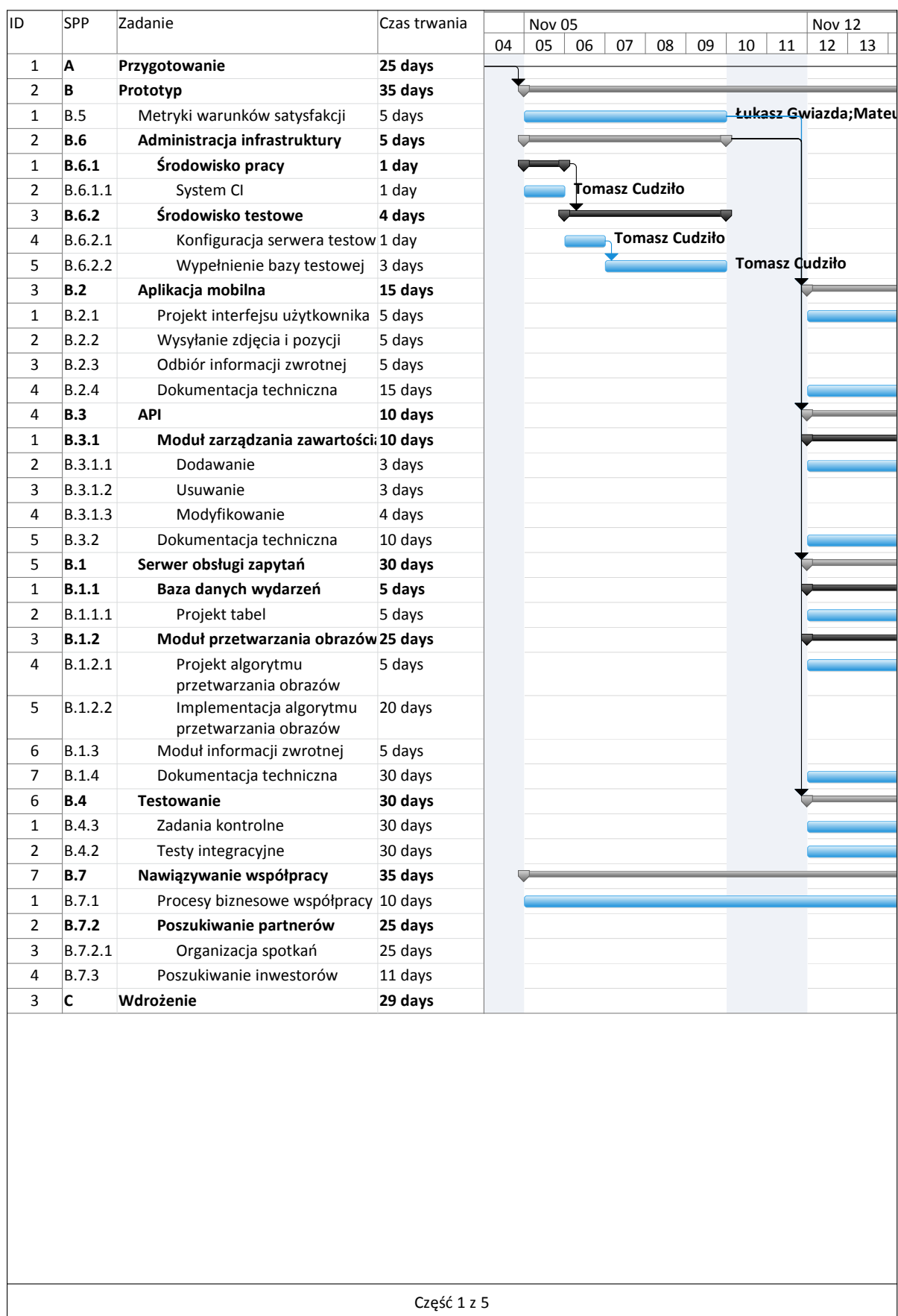


Rysunek 5: Harmonogram pracy iteracji Przygotowanie projektu Concerto. (kontynuacja)

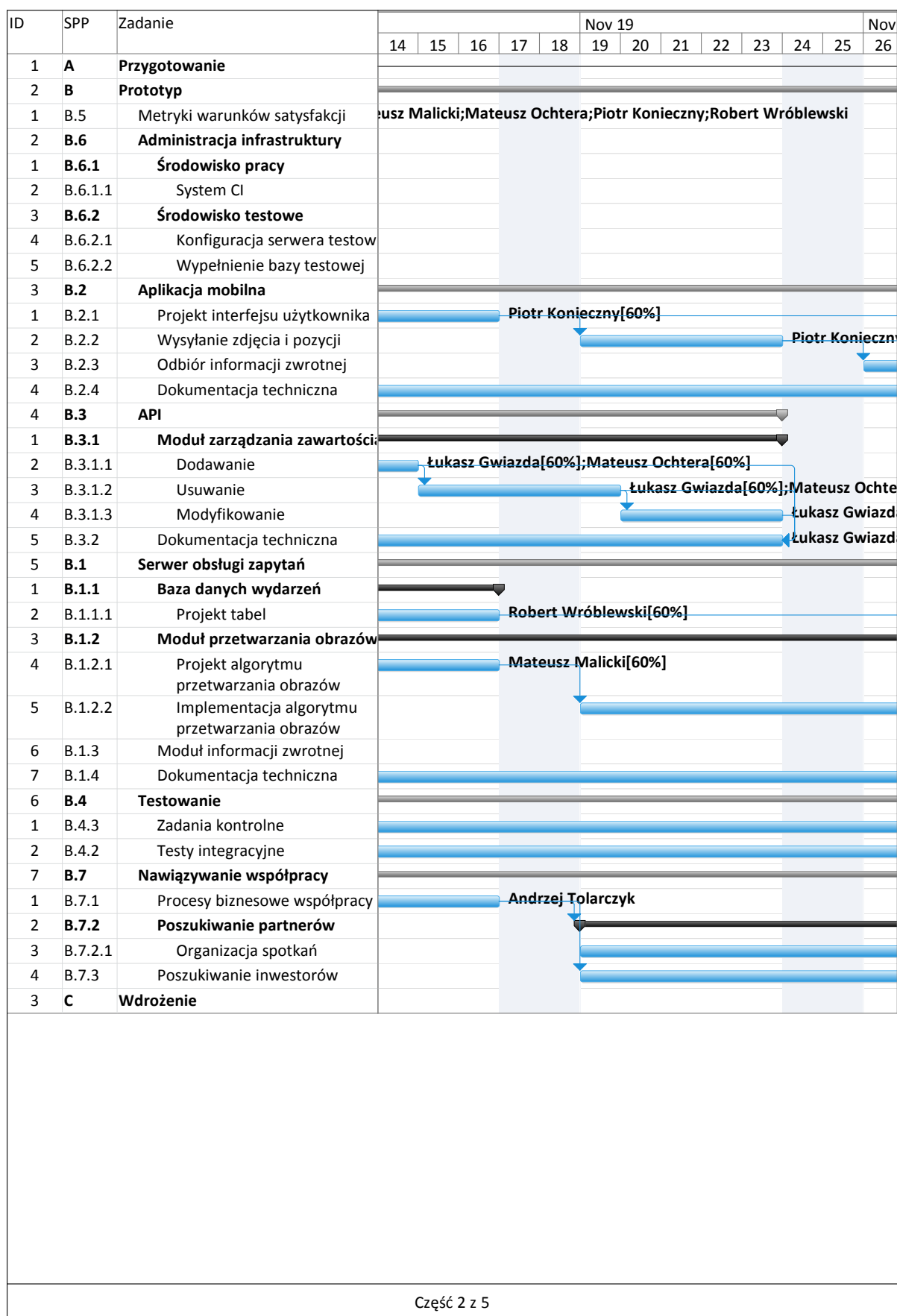


Część 4 z 4

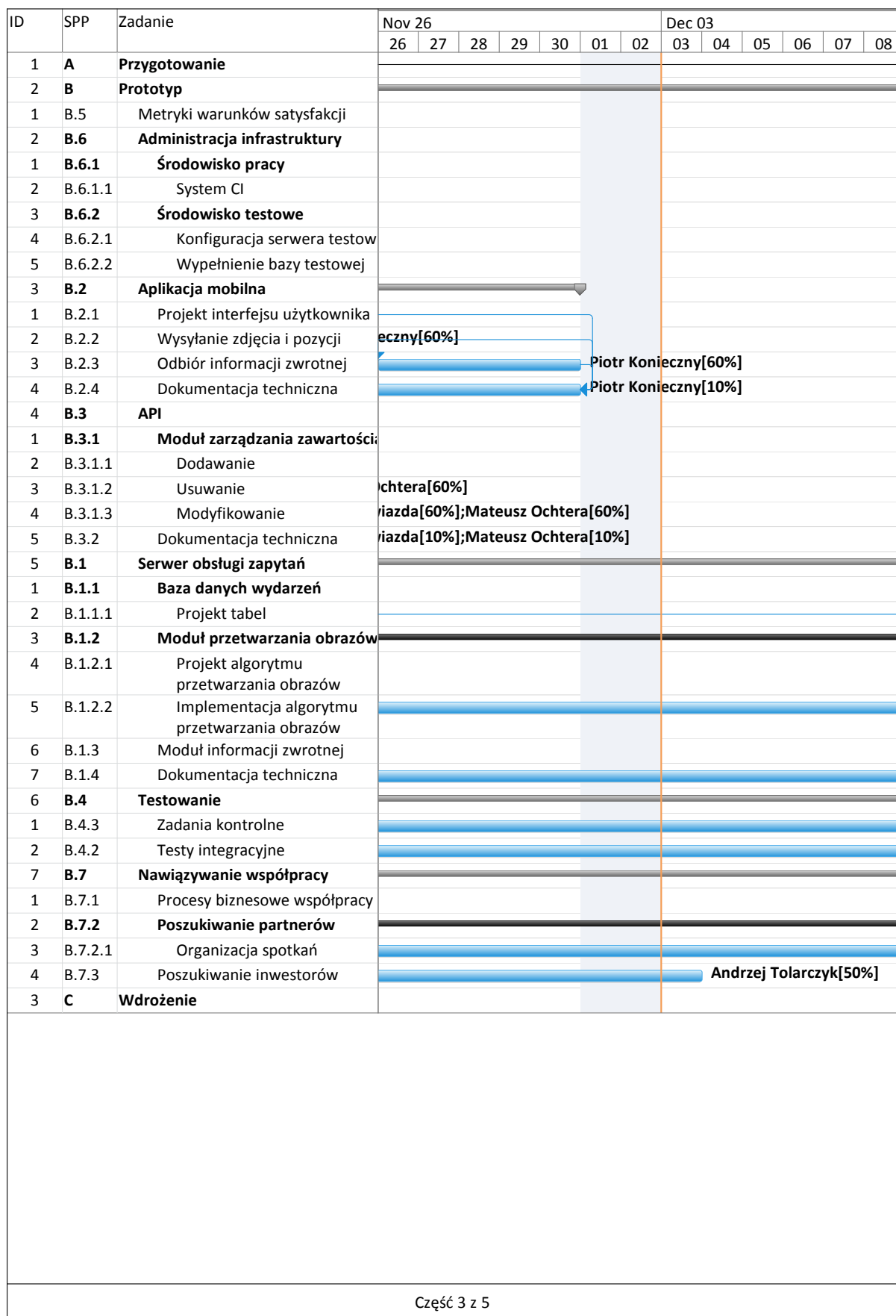
Rysunek 5: Harmonogram pracy iteracji Przygotowanie projektu Concerto. (kontynuacja)



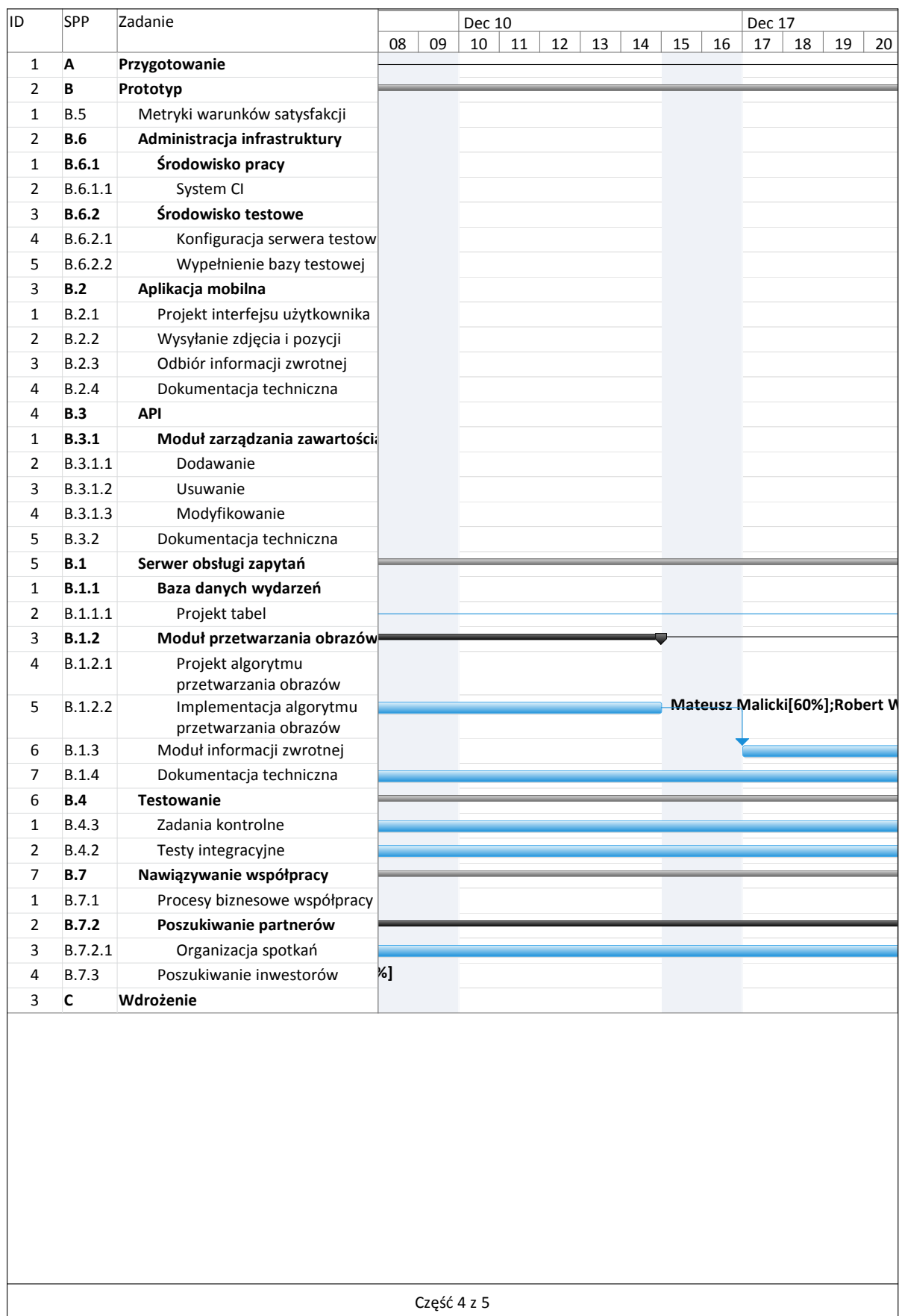
Rysunek 6: Harmonogram pracy iteracji *Prototyp* projektu *Concerto*.



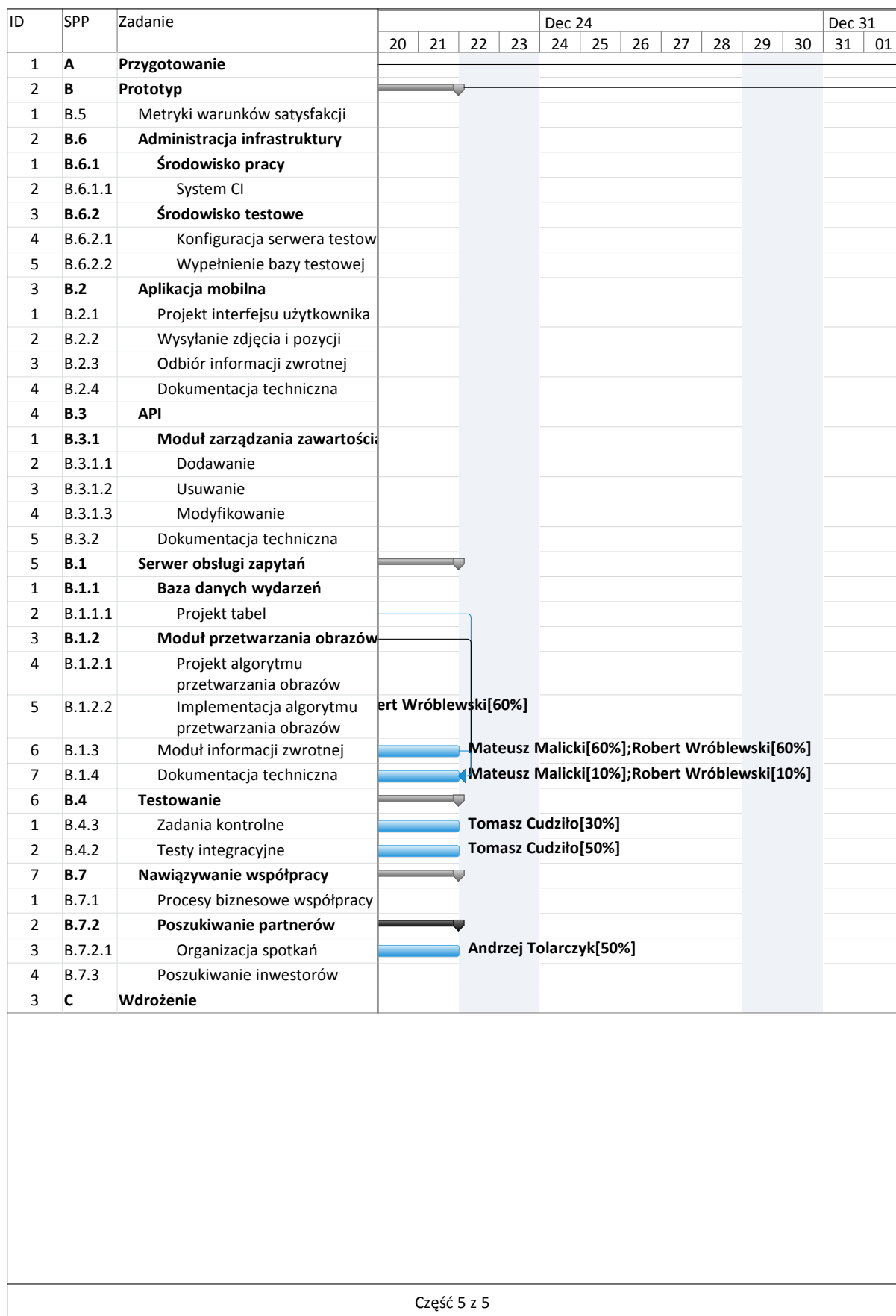
Rysunek 6: Harmonogram pracy iteracji *Prototyp* projektu *Concerto*. (kontynuacja)



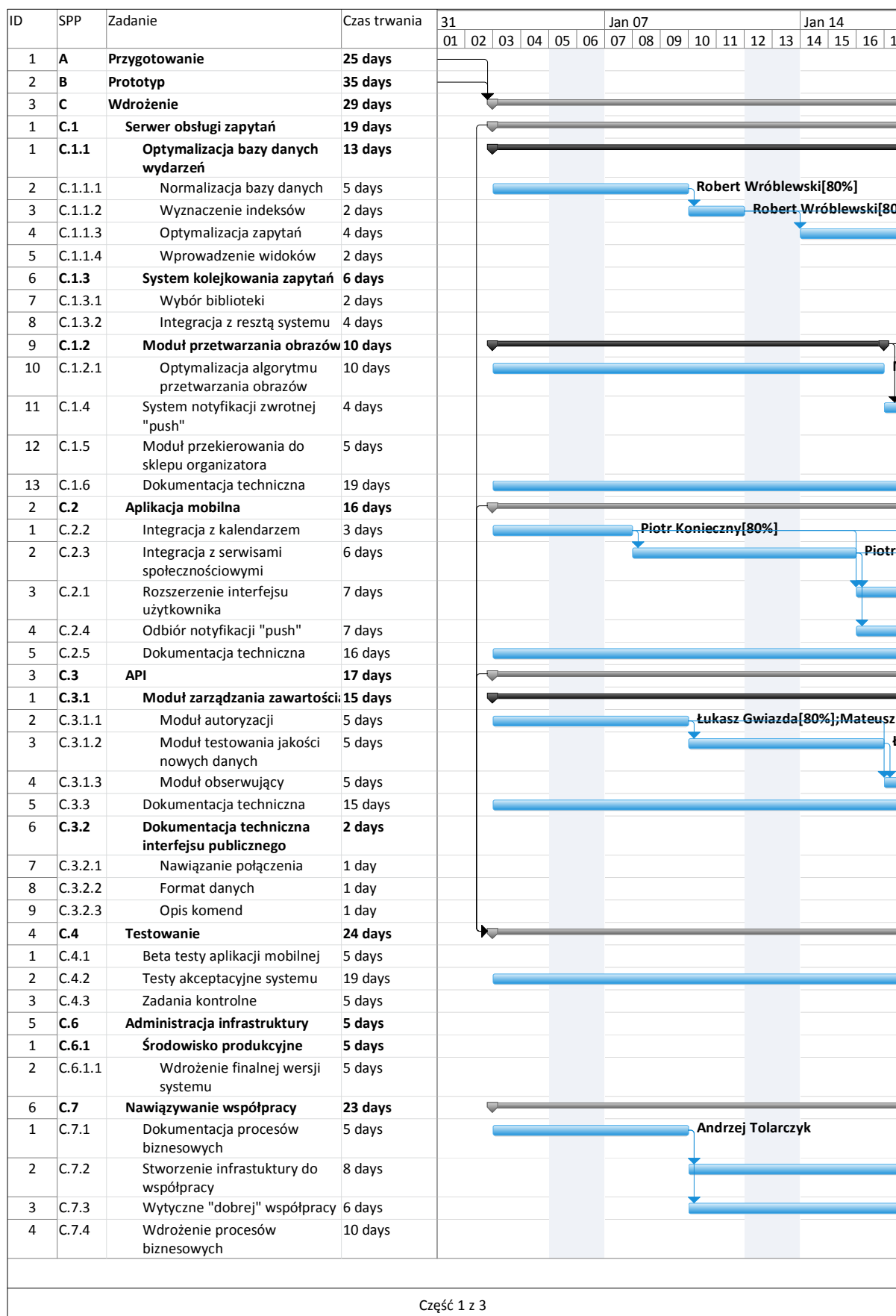
Rysunek 6: Harmonogram pracy iteracji *Prototyp* projektu *Concerto*. (kontynuacja)



Rysunek 6: Harmonogram pracy iteracji *Prototyp* projektu *Concerto*. (kontynuacja)

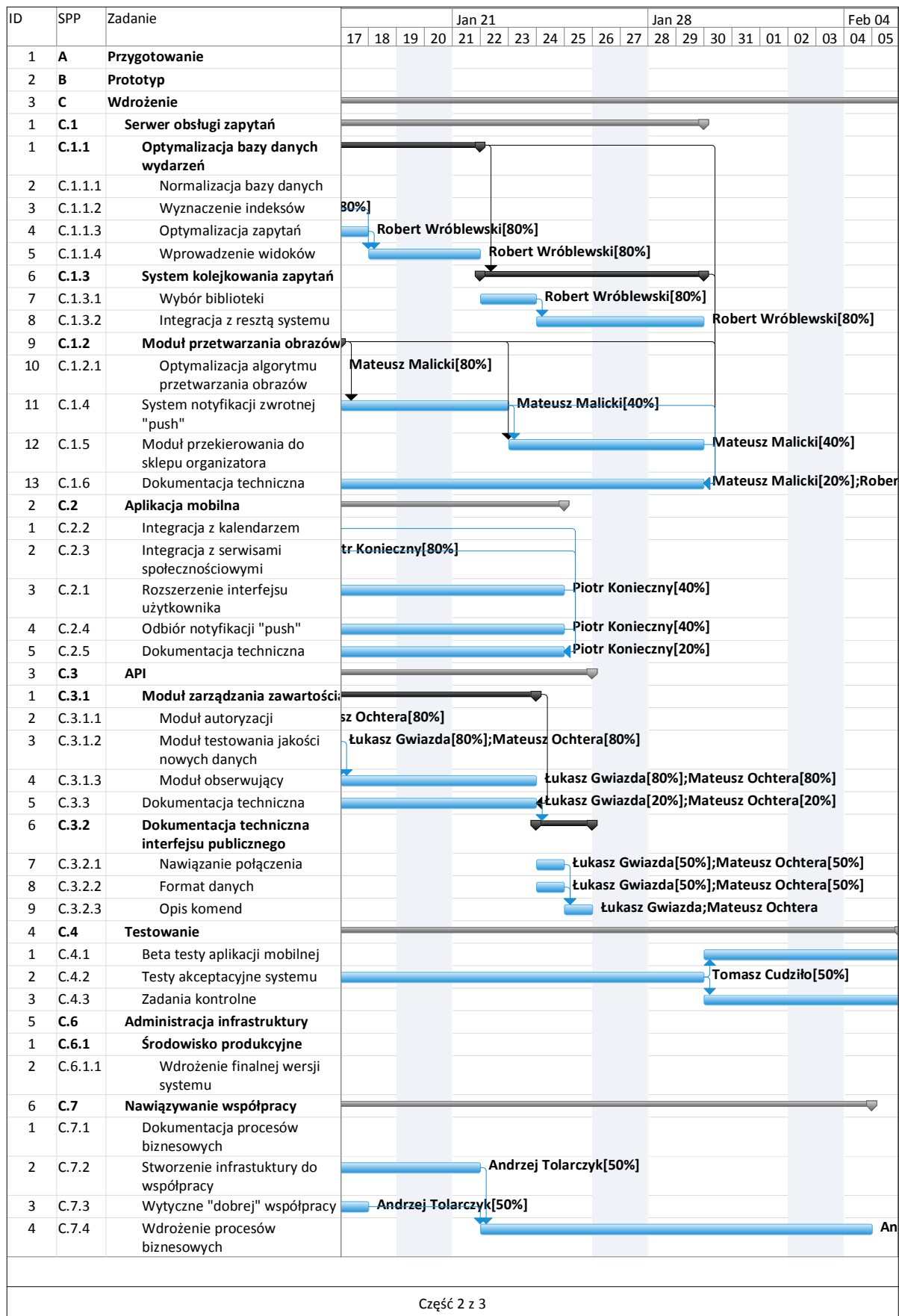


Rysunek 6: Harmonogram pracy iteracji *Prototyp* projektu *Concerto*. (kontynuacja)

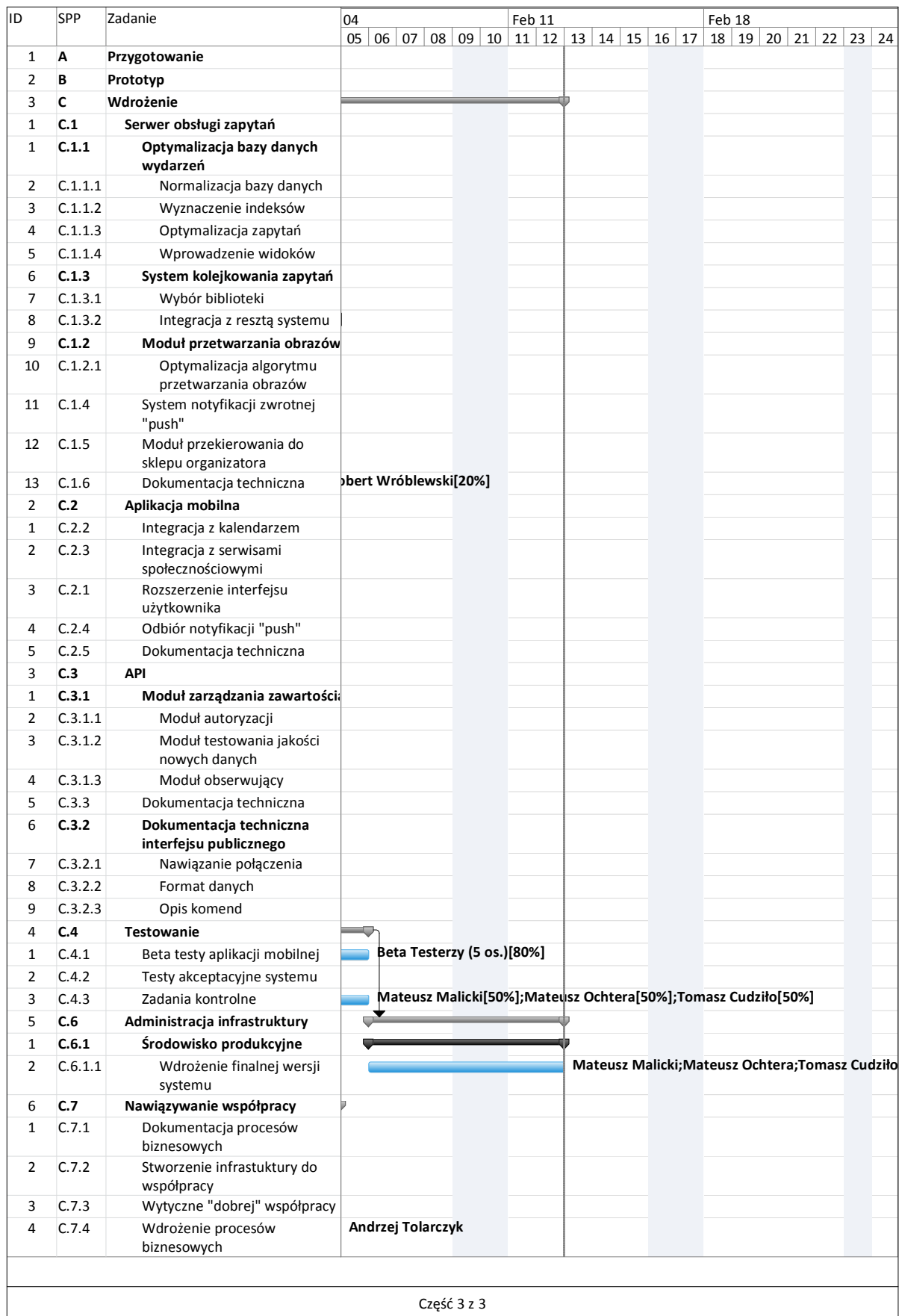


Część 1 z 3

Rysunek 7: Harmonogram pracy iteracji *Wdrożenie* projektu *Concerto*.



Rysunek 7: Harmonogram pracy iteracji Wdrożenie projektu Concerto. (kontynuacja)



Część 3 z 3

Rysunek 7: Harmonogram pracy iteracji *Wdrożenie* projektu *Concerto*. (kontynuacja)

Załączniki

Historia dokumentu

Data	Wersja	Autor	Szczegóły
2012-01-03	0.1	TC	Dodano <i>Wprowadzenie, Definicje i Zarys projektu.</i>
2013-01-03	0.2	TC	Dodano <i>Produkty, Organizację projektu, Zarządzanie oraz Harmonogram.</i>
2013-01-03	0.2.1	TC	Rozszerzono <i>Harmonogram</i> o daty iteracji, zasoby i budżet.
2013-01-03	0.3	TC	Dodano <i>Model procesu projektowego.</i>
2013-01-03	0.3.1	TC	Rozszerzono <i>Model procesu projektowego.</i>
2013-01-04	0.4	TC	Dodano <i>Proces techniczny.</i>
2013-01-04	0.4.1	TC	Dodano narzędzia do <i>Proces techniczny.</i>
2013-01-04	0.4.2	TC	Przeniesiono część techniczną opisu <i>Model procesu projektowego</i> do <i>Proces techniczny.</i>
2013-01-06	1.0	TC	Sprawdzono. Zatwierdzono.