

# Przebieg realizacji projektu wraz z opisem opracowywanych przy tym dokumentów

---

## Spis treści

Cele i zakres systemu.....	1
Określenie wymagań .....	2
Modelowanie.....	3
Projektowanie .....	7
Implementacja (programowanie).....	10
Dokumentacja .....	11
Testowanie .....	11
Wdrożenie .....	12
Konserwacja .....	13

## Cele i zakres systemu

Zwykle oprogramowanie jest tworzone dla konkretnej organizacji (np. szkoła, uczelnia, szpital, firma itd.). Dlatego analitycy pracujący w projekcie pojawiają się w tej organizacji i na początku przeprowadzają wywiady z klientami, które mają na celu:

1. dokładne zdefiniowanie problematyki projektu, czyli do czego ma służyć tworzony system komputerowy i jakie problemy mają być rozwiązywane za jego pomocą,
2. ustalenie tzw. interesariuszy projektu, czyli osób lub grup osób, które są potencjalnymi użytkownikami wytworzonego oprogramowania,
3. ustalenie jakie są gromadzone dane w organizacji i kto z użytkowników je wprowadza do systemu,
4. ustalenie rodzajów potrzebnych raportów (wydruków) które mają być generowane i udostępniane w projekcie oraz kto z użytkowników je generuje i z nich korzysta,
5. ustalenie rodzajów potrzebnych dokumentów które mają być generowane w projekcie oraz kto z użytkowników je ma generować i z nich korzystać,
6. ustalenie przepływu informacji środowisku, w którym ma pracować system,
7. ustalenie uprawnień (praw dostępu) poszczególnych grup użytkowników (np. ustawa o ochronie danych osobowych).

Później we współpracy z klientami analitycy:

- a) dokonują inwentaryzacji stanu bieżącego, czyli ustalają jaki jest aktualny stan obsługi informatycznej organizacji (czy jest system komputerowy i jak działa, a jeśli nie ma, to jak organizacja funkcjonuje - wewnętrzne procedury organizacyjne i obieg dokumentów na papierze),
- b) określają co zdaniem klienta trzeba zmienić w funkcjonowaniu organizacji, bądź w jej systemie komputerowym.

Wynikiem tej fazy ma być dokument zwany *Raportem dotyczącym celów i zakresu systemu*, który opisuje zagadnienia z tego podrozdziału, a za jego opracowanie odpowiada **kierownik projektu**. W zespole 3-osobowym kierownik jest jednocześnie głównym analitykiem, choć pozostali członkowie także starają się wczuwać w rolę analityka (np. podczas prowadzonych dyskusji w zespole), przez co wspierają kierownika. Możliwe jest także, że wybrani członkowie zespołu, na polecenie kierownika opracują fragmenty tego dokumentu.

## Określenie wymagań

Następnie, wykorzystując zdobytą wiedzę, analitycy formułują wymagania dotyczące produktu. Są to wymagania funkcjonalne, нефункционалне, sprzętowo-programowe i dotyczące baz danych.

- **Wymagania funkcjonalne** – dotyczą tego co ma realizować system; jakie ma spełniać funkcje, jakich dostarczać usług, jak zachowywać się w określonych sytuacjach. Wymagania funkcjonalne powinny być kompletne (opisywać wszystkie usługi żądane od systemu) i spójne (nie zawierać stwierdzeń sprzecznych).
- **Wymagania нефункционалне** – dotyczą tego jak system powinien realizować swoje zadania; np. wymagania dotyczące koniecznych zasobów, ograniczeń czasowych, niezawodności, bezpieczeństwa, przenośności, współpracy z określonymi narzędziami i środowiskami, zgodności z normami i standardami, a także przepisami prawnymi, w tym dotyczącymi tajności i prywatności, itp. Wymagania нефункционалне dla wielu systemów są co najmniej tak ważne jak wymagania funkcjonalne (np. szybkość działania wyszukiwarki może być równie ważna jak precyzja wyszukiwania)
- **Wymagania sprzętowo-programowe** - na jakim sprzęcie komputerowym ma działać system, pod jakim systemem operacyjnym oraz w jakim języku programowania system będzie napisany.
- **Wymagania dotyczące baz danych** - dotyczą głównie oprogramowania do budowy baz danych wykorzystywanych w projekcie (zwykle chodzi o konkretny RDBMS).

W kolejnym kroku analitycy sporządzają **opis proponowanego systemu komputerowego** w ramach projektu (jest to słowny opis działania proponowanego systemu komputerowego).

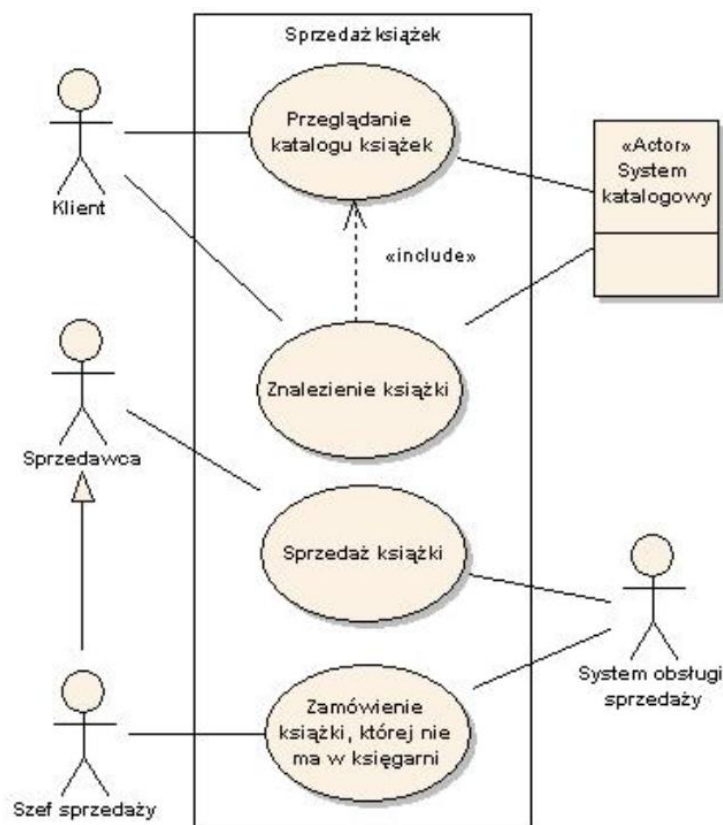
Wynikiem tej fazy ma być dokument zwany *Raportem wymagań dotyczącym tworzonego systemu*, który opisuje zagadnienia z tego podrozdziału, a za jego opracowanie odpowiada **kierownik**

**projektu.** W zespole 3-osobowym kierownik jest jednocześnie głównym analitykiem, choć pozostali członkowie także starają się wczuwać w rolę analityka (np. podczas prowadzonych dyskusji w zespole), przez co wspierają kierownika. Możliwe jest także, że wybrani członkowie zespołu, na polecenie kierownika opracują fragmenty tego dokumentu.

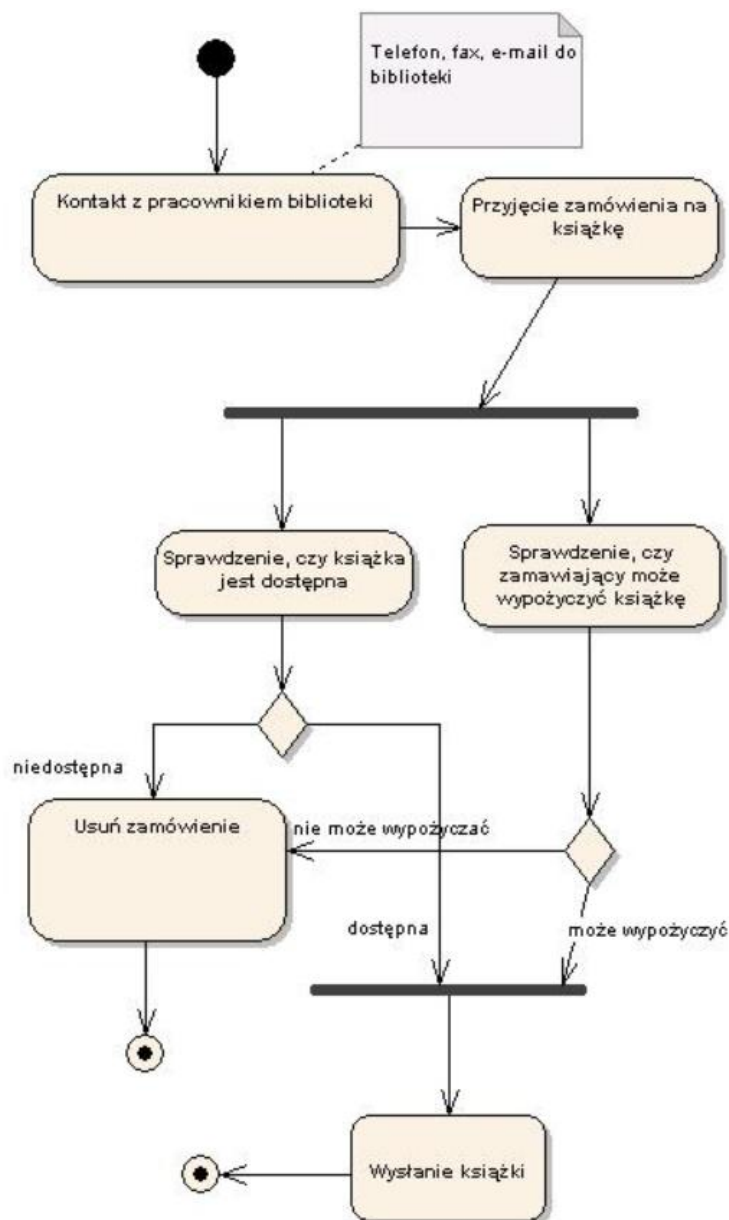
## Modelowanie

Modelowanie systemu komputerowego można wykonywać według następujących kroków.

1. Określenie tzw. granic systemu, które w podejściu UML modelowane są za pomocą aktorów. Typowym sposobem do osiągnięcia tego celu jest **diagram przypadków użycia** systemu. Jest to jeden z podstawowych diagramów UML, który definiuje interakcje jakie zachodzą pomiędzy systemem a elementami zewnętrznymi (np. użytkownikami). Może określać użytkowników systemu oraz definiuje czynności jakie mogą wykonywać. Poniżej podajemy przykład diagramu przypadków użycia dla księgarni.

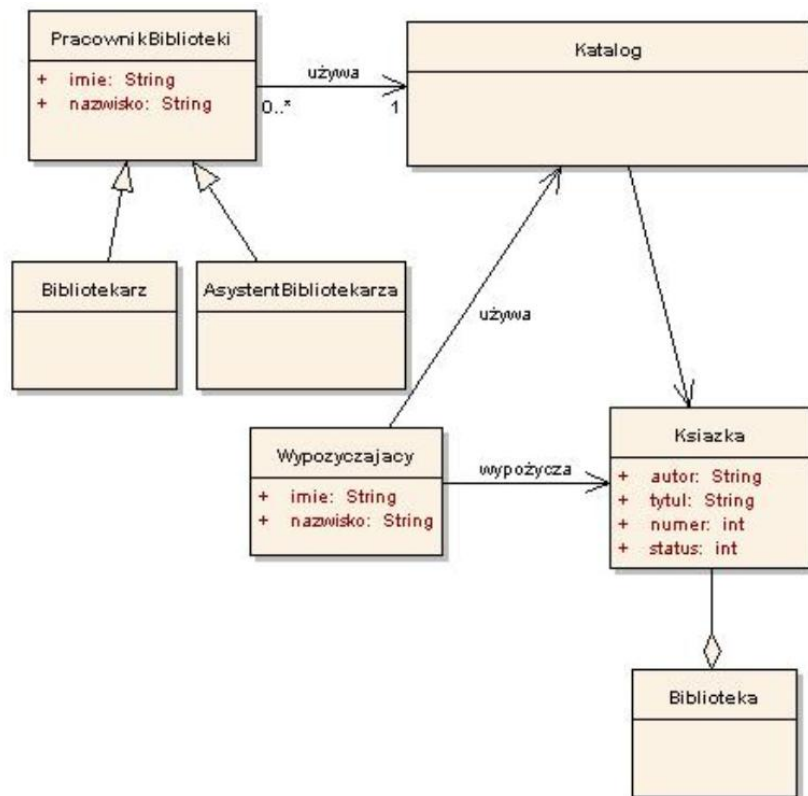


2. Określenie działań (czynności) jakie wykonują poszczególni aktorzy oraz zapisanie w jakiej kolejności te czynności są wykonywane. Typowym sposobem do osiągnięcia tego celu jest **diagram czynności**, który wykonujemy dla każdego aktora. Poniżej zamieszczono przykładowy diagram czynności (przetwarzanie zamówienia).

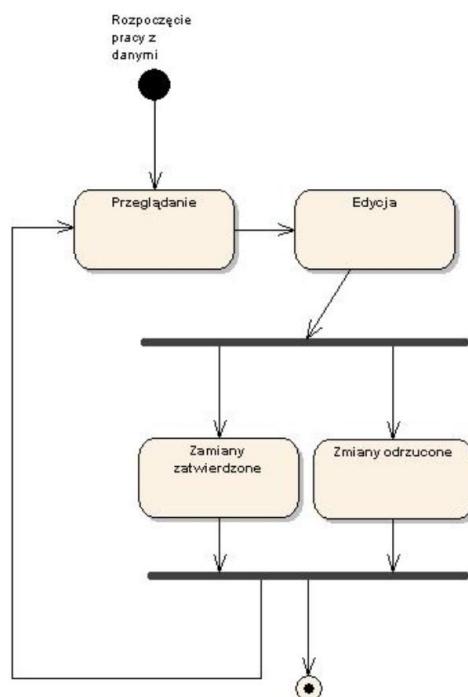


Jeśli dwóch aktorów działa jednocześnie i współpracują ze sobą, należy wykonać diagram czynności obydwu aktorów razem (np. klient i sprzedawca w sklepie).

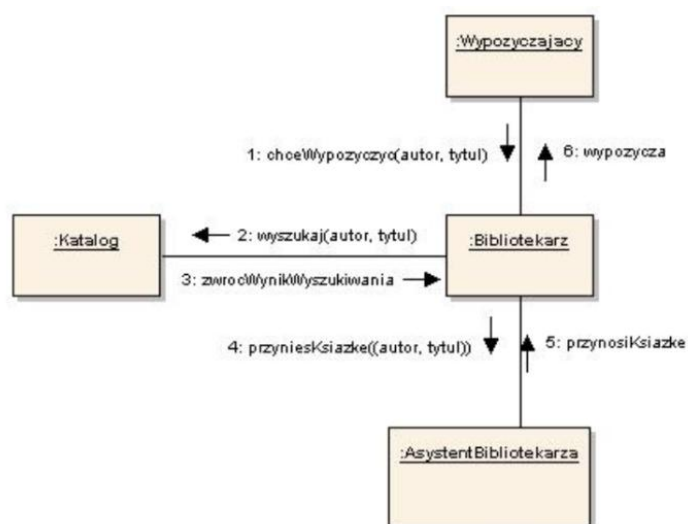
3. Kolejnym etapem modelowania jest określenie kluczowych elementów systemu, które nazywamy klasami, oraz relacji jakie między nimi występują. Typowym sposobem do osiągnięcia tego celu jest **diagram klas**, który pozwala na opisanie wewnętrznej struktury systemu, na podstawie której system zostanie następnie zaimplementowany. Klasy na schemacie należy rozumieć jako wzorniki (szablony), według których podczas działania są tworzone ich egzemplarze, czyli obiekty. Poniżej podajemy diagram klas dla systemu zarządzającego biblioteką.



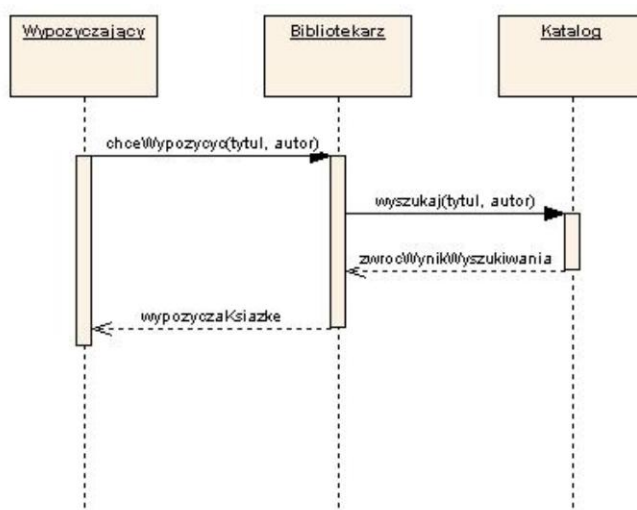
4. Po zdefiniowaniu diagramu klas, należy dla każdego obiektu będącego potencjalnym egzemplarzem klas, zamodelować **diagram stanów** tego obiektu. Diagram ten pozwala na zobrazowanie wewnętrznych stanów w jakim może znajdować się obiekt w trakcie jego użytkowania. Pozwala także zdefiniować zdarzenia, które doprowadzają do zmiany stanu modelowanego obiektu. Poniżej pokazujemy diagram stanu obiektu dane.



5. Następnym etapem jest określenie wzajemnego oddziaływania obiektów na siebie oraz komunikatów, jakie do siebie wysyłają. Jednym z dwóch typowym sposobów do osiągnięcia tego celu jest **diagram współpracy (komunikacji)**. Poniżej podajemy przykład diagramu współpracy dla biblioteki.



Natomiast drugi sposobem modelowania oddziaływania obiektów na siebie jest **diagram przebiegu (sekwencji)**. W przeciwieństwie do diagram komunikacji, który koncentruje się na zobrazowaniu współpracy między obiektami, diagram przebiegu pokazuje kolejność przesyłania komunikatów i czas istnienia obiektów. Poniżej podajemy przykład diagramu sekwencji dla biblioteki.



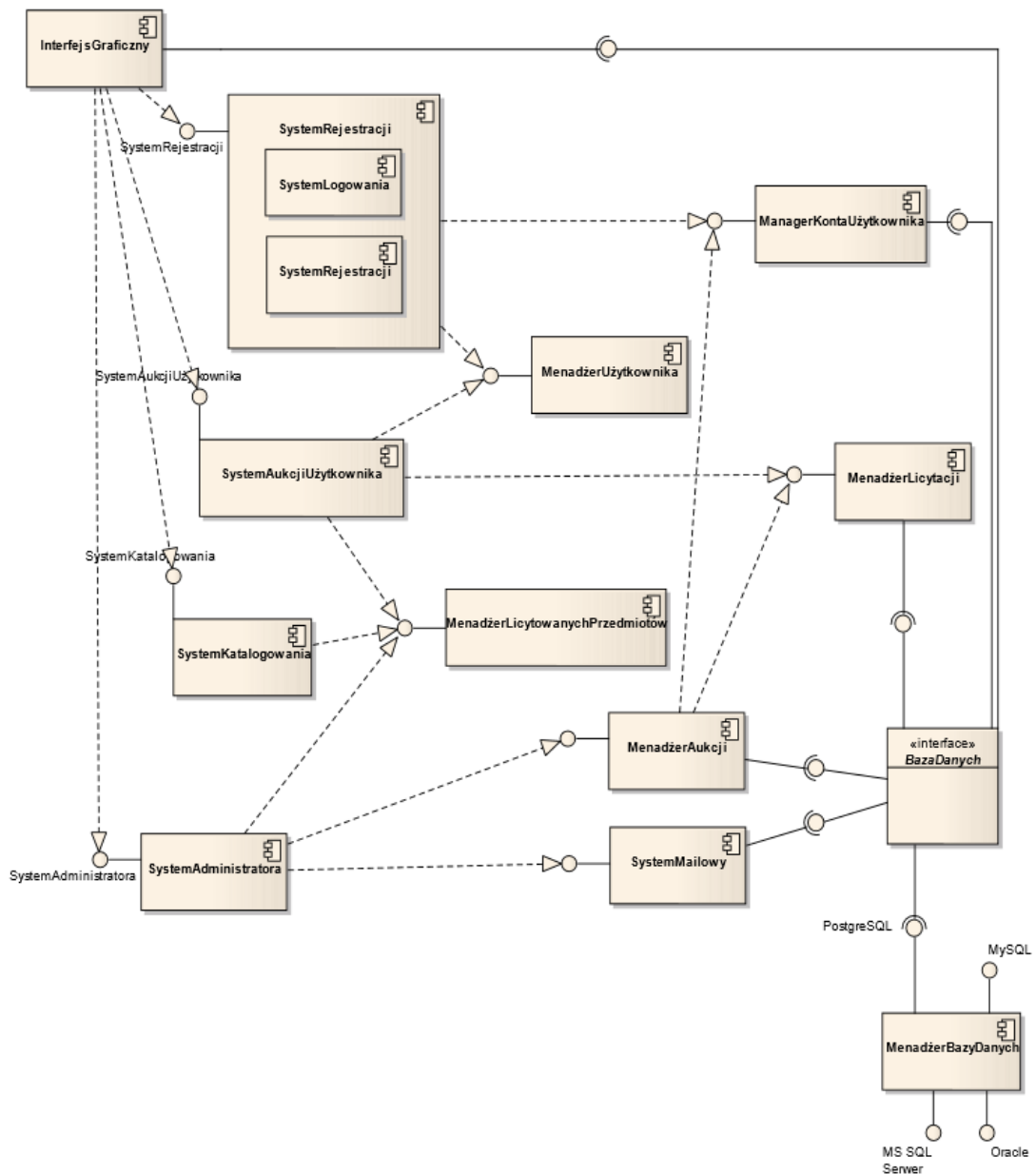
Na diagramach komunikacji i sekwencji aktorzy są zwykle modelowani za pomocą obiektów je reprezentujących.

Wynikiem tej fazy ma być dokument zwany *Raportem modelowania systemu*, który opisuje zagadnienia z tego podrozdziału, a za jego opracowanie odpowiada **kierownik projektu**. W zespole 3-osobowym kierownik jest jednocześnie głównym analitykiem, choć pozostali członkowie także starają się wczuwać w rolę analityka (np. podczas prowadzonych dyskusji w zespole), przez co wspierają kierownika. Możliwe jest także, że wybrani członkowie zespołu, na polecenie kierownika opracują fragmenty tego dokumentu.

## Projektowanie

Celem fazy projektowania jest opracowanie szczegółowego planu implementacji systemu. W odróżnieniu od analizy, w projektowaniu dużą rolę odgrywa środowisko implementacji. Projektanci muszą posiadać dobrą znajomość języków, bibliotek i narzędzi stosowanych w trakcie implementacji.

Projektowanie można rozpocząć od wykonania schematu architektury systemu, którym zostanie odzwierciedlony podział na komponenty systemu. Schemat pokaże także zależności pomiędzy modułami. Typowym sposobem do osiągnięcia tego celu jest **diagram komponentów**. Diagramy komponentów pokazują podział systemów na mniejsze podsystemy. Komponent to wymienny, wykonywalny fragment systemu o hermetyzowanych szczegółach implementacyjnych. Funkcjonalność oferowana przez komponent jest dostępna przez interfejsy, które implementuje. Z drugiej strony, komponent może wymagać pewnych interfejsów, które muszą być dostarczone przez inne komponenty. Diagram komponentów służy zatem do pokazania związków pomiędzy komponentami i interfejsami



Następnie wykonywany jest projekt każdego modułu poprzez sporządzenie szczegółowego diagramu klas dla tego modułu (jest to uszczegółowienie diagramu klas z etapu modelowania).

Ponadto, w przypadku modułu będącego bazą danych, baza danych jest projektowana według poniższych kroków:

- formułowanie definicji celu oraz założeń wstępnych bazy danych (opis słowny),
- tworzenia struktur danych (diagram ERD bez powiązań),
- definiowanie powiązań między tabelami (kompletny diagram ERD)
- określenie reguł integralności (opis słowny),
- definiowanie perspektyw (opis słowny).

Natomiast w przypadku modułów GUI, wykonywane są makiety głównych okien dialogowych modułów.



Następnie projektowane są narzędzia do komunikacji pomiędzy modułami. Mogą to być klasy komunikacyjne, gdzie są specjalne metody realizujące komunikację.

Wynikiem tej fazy powinna być lista zadań do implementacji.

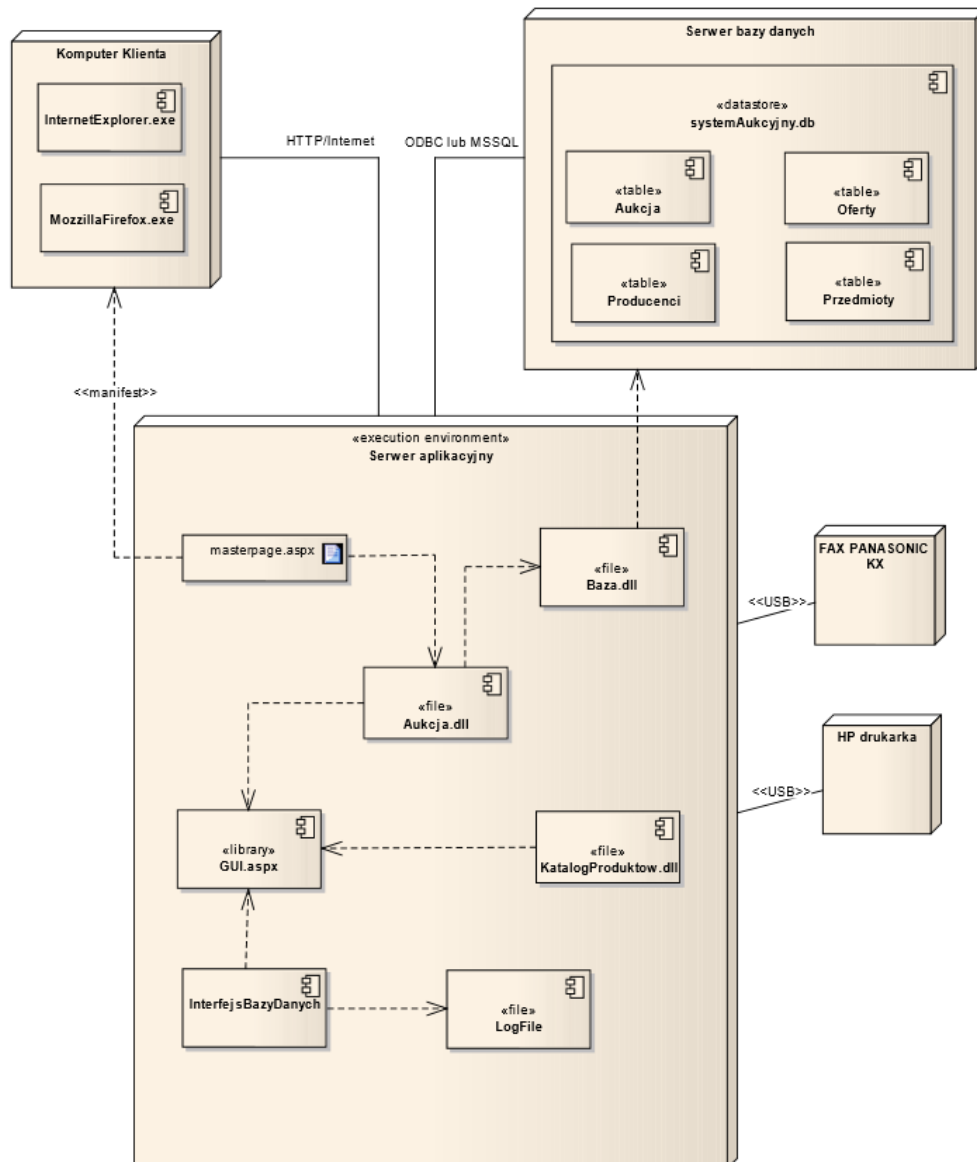
Na końcu tej fazy pojawia się także szczegółowy harmonogram implementacji projektu.

Harmonogram określa rozbić całości pracy na poszczególne zadania wraz z ustaleniem czasu realizacji zadań (oraz ewentualnie kosztem realizacji, przydzielonymi zasobami i efektem końcowym każdego pojedynczego zadania). Pomocne przy układaniu harmonogramu są rozmaite sposoby ilustracji, jak np. tzw. diagramy Gantta.

ZADANIE	Styczeń																															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
Zadanie 1																																
Zadanie 2																																
Zadanie 3																																
Zadanie 4																																
Zadanie 5																																

Można to także robić w tabelce wordowej (w załączniku).

Dla nietrywialnie prostych systemów wykonuje się także **diagram wdrożenia** przedstawiający powiązania między **oprogramowaniem i sprzętem**. Taki odzwierciedla fizyczną strukturę całego systemu, z uwzględnieniem oprogramowania i sprzętu. Jednostki oprogramowania są reprezentowane przez komponenty uruchomieniowe oraz bazy danych i biblioteki. Stronę sprzętową reprezentują węzły, czyli poszczególne urządzenia obliczeniowe i komunikacyjne.



Wynikiem tej fazy ma być dokument zwany *Raportem z projektowania systemu*, który opisuje zagadnienia z tego podrozdziału, a za jego opracowanie odpowiada **kierownik projektu**. W zespole 3-osobowym kierownik jest jednocześnie głównym projektantem, choć pozostali członkowie także starają się wczuwać w rolę projektanta (np. podczas prowadzonych dyskusji w zespole), przez co wspierają kierownika. Możliwe jest także, że wybrani członkowie zespołu, na polecenie kierownika opracują fragmenty tego dokumentu.

## Implementacja (programowanie)

Implementacja polega na zaprogramowaniu wszystkich modułów systemu, w tym zaimplementowaniu bazy danych oraz modułów GUI.

Jako główne kroki implementacji bazy danych, zwykle wykonywane są:

- skonstruowanie i wykonanie odpowiednich poleceń tworzących tabele,
- przygotowanie oraz wstawienie początkowych danych do tabel,
- sformułowanie i zaimplementowanie wszystkich potrzebnych zapytań do bazy danych,
- skonstruowanie i wykonanie odpowiednich poleceń tworzących perspektywy,
- utworzenie grup użytkowników w bazie danych oraz zdefiniowanie uprawnień tych grup,
- utworzenie początkowych użytkowników w poszczególnych grupach.

Dla każdego modułu, oprócz implementacji samego modułu, implementowane są także biało-skrzynkowe testy jednostkowe tego modułu.

Następnie następuje integracja modułów oraz tworzone są biało-skrzynkowe testy integracyjne.

Powstaje także opis testów jednostkowych i integracyjnych.

Wynikiem tej fazy jest oczywiście powstałe oprogramowanie. Ponadto każdy programistów tworzących moduły opracowuje dokumentację modułu (JavaDoc) oraz dokument zwany *Opisem testów biało-skrzynkowych systemu*, który opisuje jakie testy jednostkowe i integracyjne zostały zaimplementowane. W zespole 3-osobowym kierownik może pełnić rolę programisty, ale większość prac programistycznych powinien wykonywać drugi członek zespołu. Oczywiście kierownik nadzoruje wykonanie tych prac i odpowiada za ich wykonanie.

## Dokumentacja

W skład dokumentacji końcowej (dostarczanej odbiorcom) najczęściej wchodzi:

- podręcznik użytkownika (user's guide)
- dokumentacja techniczna (reference manual)

Szczegóły w osobnym dokumencie.

Wynikiem tej fazy jest oczywiście powstała dokumentacja. W zespole 3-osobowym kierownik może pomagać w pracach dokumentacyjnych, ale większość tych prac powinien wykonywać trzeci członek zespołu. Oczywiście kierownik nadzoruje wykonanie tych prac i odpowiada za ich wykonanie.

## Testowanie

Ogólnie, testy dzielą się na biało-skrzynkowe i czarno-skrzynkowe.

Testy biało-skrzynkowe (często wykonywane bez nadzoru np. przez serwer ciągłej integracji) mogą być to testy jednostkowe dla modułów i integracyjne dla całego systemu. Na końcu etapu testowania wszystkie testy powinny przejść bez błędów.

Testy czarno-skrzynkowe polegają na wykonywaniu przez testerów ustalonych scenariuszy wykorzystywania systemu. Testowana może być zarówno poprawność, jak i efektywność działania. Przykłady scenariuszy do testowania powinny zostać specjalnie zaprojektowane, aby przetestować wszystkie funkcjonalności. Ponadto, do scenariuszy trzeba przygotować właściwe dane.

Proces testowania powinien być zaplanowany w ramach całego procesu wytwarzania oprogramowania (kiedy i jakie testy będą przeprowadzane).

Oczekiwane wyniki testów powinny być ściśle określone. Powinny być także określone procedury postępowania stosownie do wyników testów.

Całość testów (zestaw wszystkich przykładów) i wyników powinny być odpowiednio udokumentowane.

Wynikiem tej fazy ma być dokument zwany *Raportem z testów wytworzonego oprogramowania*, który opisuje wykonane testy oraz ich wyniki. W zespole 3-osobowym kierownik może pomagać w testach, ale większość tych prac powinien wykonywać trzeci członek zespołu (ten, który wykonał dokumentację). Oczywiście kierownik nadzoruje wykonanie tych prac i odpowiada za ich wykonanie.

## Wdrożenie

Wdrożenie oprogramowania polega na przygotowaniu do użycia przez klienta konkretnej konfiguracji oprogramowania dla konkretnego środowiska sprzętowego. Oprogramowanie powinno być w formie „wydania” (release), zawierającego program oraz ewentualnie elementy dodatkowe: pliki konfiguracyjne, pliki danych do przykładów i dokumentację.

Wdrożenie obejmuje dostarczenie oprogramowania i (opcjonalnie) sprzętu, skonfigurowanie sprzętu i oprogramowania, uruchomienie systemu. Wdrożenie może być dokonywane przez klienta (zazwyczaj przy użyciu odpowiednich narzędzi), niezależną od producenta firmę, wreszcie przez przedstawicieli producenta.

Podsumowując, wdrożenie obejmuje następujące działania.

- Szkolenie użytkowników
- Instalacja sprzętu i oprogramowania
- Wypełnienie baz danych przykładowymi danymi.
- Nadzorowane korzystanie z systemu
- Usuwanie błędów w oprogramowaniu.
- Przekazanie systemu klientowi.

Wynikiem tej fazy ma być wdrożony system. W zespole 3-osobowym wdrożeniem systemu zajmuje się drugi członek zespołu (główny programista).

## Konserwacja

Typowe czynności związane z konserwacją obejmują:

- przyjmowanie zgłoszeń o błędach i propozycji zmian w funkcjonowaniu systemu,
- oszacowanie koniecznych zmian w kodzie,
- zaplanowanie harmonogramu wprowadzania zmian,
- wprowadzenie zmian,
- przygotowanie poprawek lub kolejnego wydania.

Wynikiem tej fazy ma być krótki dokument opisujący planowaną konserwację systemu. Dokument ten przygotowuje kierownik projektu.