

Zagadnienia do egzaminu z Podstaw Inżynierii Oprogramowania

1. Modele procesu wytwarzania oprogramowania.
Przegląd modeli cyklu życia oprogramowania. Model kaskadowy. Realizacja kierowana dokumentami. Prototypowanie. Realizacja przyrostowa. Montaż z gotowych elementów. Model spiralny. Formalne transformacje.
2. Fazy strategiczna oraz określania wymagań.
Faza strategiczna jako studium wykonalności. Ocena rozwiązań. Szacowanie kosztów oprogramowania. Algorytmiczne modele szacowania kosztów oprogramowania. Faza określania wymagań. Wymagania stawiane oprogramowaniu. Proces inżynierii wymagań. Wymagania funkcjonalne, wymagania нефункционалне, dokumentowanie wymagań.
3. Faza analizy – modelowanie strukturalne.
Faza modelowania dziedziny problemu: strukturalne metody analizy. Model dziedziny problemu, modelowanie strukturalne – omówienie metodyk i systemów notacji, modele funkcji, modele danych. Diagramy związków encji. Diagramy przepływów danych. Diagramy przejść stanów. Specyfikacja modelu strukturalnego. Proces tworzenia modelu strukturalnego. Modelowanie danych. Modelowanie procesów i przepływów danych. Modelowanie przejść stanów.
4. Faza analizy – modelowanie obiektowe.
Faza modelowania dziedziny problemu: obiektowe metody analizy. Język obiektowego modelowania UML. Perspektywy modelu obiektowego i rodzaje diagramów w notacji UML. Diagram przypadków użycia – elementy notacji, analiza aktorów i przypadków użycia. Notacje obiektowe i ich interpretacja. Diagramy klas i obiektów. Diagramy interakcji. Diagramy przejść stanów. Specyfikacja modelu obiektowego. Proces tworzenia modelu obiektowego. Budowa statycznego modelu klas. Identyfikacja klas i obiektów. Identyfikacja związków klas i obiektów. Identyfikacja i definiowanie pól. Identyfikacja i definiowanie metod i komunikatów. Analiza scenariuszy (przypadków użycia). Modelowanie przejść stanów.
5. Projektowanie oprogramowania.
Uszczegółowienie wyników analizy: techniki obiektowe i strukturalne. Projektowanie składowych systemu nie związanych z dziedziną problemu. Projektowanie składowej kontaktu z użytkownikiem. Projektowanie składowej zarządzania danymi. Optymalizacja projektu. Dostosowanie do ograniczeń i możliwości środowiska implementacji. Określenie fizycznej struktury systemu. Poprawność projektu. Jakość projektu, spójność oraz stopień powiązań składowych.
6. Faza implementacji.
Programowanie niezawodnego oprogramowania, unikanie błędów, tolerancja błędów. Charakterystyka typowych środowisk implementacji: języki proceduralne i obiektowe, relacyjne bazy danych, obiektowe bazy danych, środowiska programistyczne programów użytkowych, narzędzia szybkiego wytwarzania aplikacji.
7. Testowanie, weryfikacja, walidacja.
Poziomy testowania (jednostkowe, integracyjne, systemowe, akceptacja). Statyczne techniki testowania. Dynamiczne techniki testowania: strategie funkcjonalne i strukturalne. Niezawodność i bezpieczeństwo oprogramowania. Testy statystyczne. Wykrywanie błędów. Testy funkcjonalne. Testy strukturalne. Testy statyczne. Ocena liczby błędów. Testy systemu. Bezpieczeństwo oprogramowania.
8. Instalacja i konserwacja oprogramowania
Zadania wykonywane w fazie instalacji. Podstawowe rezultaty fazy instalacji. Narzędzia CASE w fazie instalacji. Modyfikowanie oprogramowania. Inżynieria odwrotna. Podstawowe rezultaty fazy konserwacji. Narzędzia CASE w fazie konserwacji.
9. Programowanie ekstremalne.
Zagadnienia: Programowanie ekstremalne (ang. eXtreme Programming, XP) jako paradygmat i metodyka programowania mające na celu wydajne tworzenie małych i średnich "projektów wysokiego ryzyka". Synergia podstawą ekstremalnego programowania. Iteracyjność. Testy podzespołów. Ciągłe modyfikacje architektury. Programowanie parami. Stały kontakt z klientem. Brak dokładnej specyfikacji. Wspólna "własność" kodu.

Wykaz źródeł (podręczników i skryptów)

1. Jaskiewicz A., Inżynieria oprogramowania, wyd. Helion 1997.
2. Ian Sommerville, Inżynieria oprogramowania, WNT, 2003.
3. Górski J. (red.), Inżynieria oprogramowania w projekcie informatycznym, wyd. Mikom 1999.
4. Yourdon E., Współczesna analiza strukturalna, WNT 1996.
5. Roszkowski J., Analiza i projektowanie strukturalne, wyd. Helion, 2002.
6. Dumnicki R., Kasprzyk A., Kozłowski M., Analiza i projektowanie obiektowe, wyd. Helion 1998.
7. Steve Maguire, Niezawodność oprogramowania, wyd. Helion 2002.