

Objektorientierter Programmentwurf nach der Unified Modelling Language (UML)

Mit der Objektorientierung soll die Komplexität der Software - Entwicklung erleichtert werden. Objekte sind geschlossene, kleine Programmeinheiten, die in Beziehung zu anderen Objekten stehen. Im objektorientierten Programmentwurf kommt es darauf an, die Objekte möglichst allgemeingültig (für Programme wiederverwendbar) definieren zu können und die Beziehungen der Objekte untereinander zu bestimmen.

Objektorientierte Programme basieren auf vier Grundsätzen:

1. Kapselung
2. Vererbung
3. Verbergen von Daten
4. Polymorphie

In der objektorientierten Programmierung unterscheidet man zwischen Klasse und Objekt. Die Klasse ist der Bauplan, in dem beschrieben wird, welche Eigenschaften (Attribute) und Operationen (Methoden) Objekte dieser Klasse besitzen. Objekte sind konkrete Exemplare (Instanzen) einer Klasse.

Klasse

Mitarbeiter
-Personalnummer : int -Abteilung : wchar_t -Konto : int
+PersonalNummerAnzeigen() : void

Objekt

Hanst_Muster.Mitarbeiter
-30264 -IT14 -123456789
+PersonalNummerAnzeigen() : void

Auf die Daten (Attribute) eines Objekts kann nur über definierte Schnittstellenfunktionen (Methoden) zugegriffen werden (Kapselung und Verbergen von Daten!).

1. Grundlegende Projektzyklen eines „Spiralmodells“

Ein objektorientiert durchgeführtes Softwareprojekt wird in unterschiedlichen Entwicklungsschritten (Projektzyklen) durchlaufen. Für jeden dieser Schritte stellt die UML Vorgehensweisen und Entwurfsmodelle zur Verfügung. 'Jeder nachfolgend dargestellte Projektzyklus kann mehrfach durchlaufen werden.

Das Projektmanagement sollte nach den Methoden der Teamarbeit realisiert werden.

Design

Basismodell:

Im Basismodell werden die Grundelemente Klassen, Objekte, Attribute und Operationen entworfen.

Statisches Modell:

Dieses Modell beschreibt die Beziehungen der Klassen und Objekte, die Ergebnis des Basismodells waren.

Dynamisches Modell:

Das dynamische Modell beschreibt den Zustand und die Reaktion der Objekte auf Aktivitäten zur Laufzeit. Die Beschreibung kann in der UML in unterschiedlichen Diagrammen erfolgen. Diese sind das Sequenz-, Kollaboration- Zustands- oder Aktivitätsdiagramm.

Realisierung

Kodierung:

Die Kodierung sollte in einer objektorientierten Programmiersprache erfolgen, da der Entwurfsansatz so direkt programmtechnisch umsetzbar ist. Beispiele sind C++, Java, Smalltalk, C#, VB.NET oder die Scriptsprache TCL.

Softwaretest:

Beim Softwaretest wird kontrolliert, ob die Ergebnisse der Designphase entsprechend umgesetzt wurden.

Einsatz

Installation:

Das Produkt wird auf den Rechnern des Auftraggebers implementiert.

Schulung:

Die Mitarbeiter erhalten eine Einweisung, um mit dem Produkt arbeiten zu können.

Review:

Ein Vergleich des tatsächlichen Endproduktes mit dem geplanten Produkt zeigt, ob Änderungen notwendig sind.