

Modulidentifikation

Modulnummer	318	
Titel	Analysieren und objektbasiert programmieren mit Komponenten	
Kompetenz	Eine Aufgabe analysieren mit einer Entwicklungsumgebung mit integriertem GUI-Designer implementieren, dokumentieren und testen.	
Handlungsziele	1.	Problemstellung analysieren, Benutzerschnittstelle entwerfen und Abläufe darstellen.
	2.	Grafische Benutzerschnittstelle gestalten und mit GUI-Designer realisieren.
	3.	Programmentwurf mit einer auf Komponenten basierenden Programmierumgebung mit Einsatz von Ablaufstrukturelementen, eigenen Prozeduren und Funktionen implementieren.
	4.	Beim Programmieren vorgegebene Codekonventionen einhalten, den Programmcode kommentieren und dabei auf Wartbarkeit und Nachvollziehbarkeit achten.
	5.	Ausgehend von der Aufgabenstellung Testfälle erstellen, das Programm testen und Fehler beheben.
Kompetenzfeld	Application Engineering	
Objekt	Mit einer Komponenten-Bibliothek implementiertes Programm	
Niveau	2	
Voraussetzungen	Objektbasiert programmieren nach Vorgabe	
Anzahl Lektionen	40	
Anerkennung	Eidg. Fähigkeitszeugnis Informatiker-in	
Modulversion		
MBK Release	R6	
Überarbeitet am	10.04.2014	
	Fokus auf Komponenten und GUI-Designer. Terminologie aktualisiert. Hinweis: Inhaltliche Überschneidung mit Modul 404	

Handlungsnotwendige Kenntnisse

Modulnummer	318
Titel	Analysieren und objektbasiert programmieren mit Komponenten
Kompetenzfeld	Application Engineering
Modulversion	3.0
MBK Release	R6

Handlungsziel	Handlungsnotwendige Kenntnisse	
1.	1.	Kennt eine Methode für die Analyse einer Problemstellung (z.B. Use Case) und kann diese auf einfache Aufgaben anwenden.
	2.	Kennt eine Methode für die Darstellung von Programmabläufen (z.B. Struktogramm, Aktivitätsdiagramm) und kann damit einfache Sachverhalte darstellen.
	3.	Kennt die Strukturelemente Iteration und Selektion und kann deren sinnigerechte Anwendung aufzeigen.
2.	1.	Kennt allgemeine Regeln der SW-Ergonomie (z.B. DIN EN ISO 9241-110) und kann aufzeigen, wie diese für die Gestaltung einer Oberfläche anzuwenden sind.
	2.	Kennt die wichtigsten Komponenten einer graphischen Oberfläche und kann diese korrekt einsetzen.
3.	1.	Kennt eine Komponenten-basierte Programmierumgebung (z.B. .NET, Java/Swing) und deren typische Einsatzgebiete.
	2.	Kennt die unterschiedlichen Möglichkeiten für Selektion (einfach/mehrfach) und Iteration (vor-/nachprüfend) und kann diese mit einer Programmiersprache umsetzen.
	3.	Kennt Prozeduren und Funktionen und kann deren Aufbau (inkl. Parameterlisten) und Einsatz erklären.
4.	4.	Kennt den Mechanismus der Ereignisverarbeitung eines objektbasierten Systems und kann Ereignisse in Verarbeitungsroutinen auswerten.
	1.	Kennt die Bedeutung von Kommentaren bei Prozeduren und Funktionen und deren Beitrag zu besser wartbarem Programmcode.
5.	1.	Kennt ein Verfahren für den statischen Test (z.B. Codereview) von Programmcode und kann die Bedeutung für die Qualität der Software aufzeigen.
	2.	Kennt ein Verfahren (z.B. Grenzwertanalyse) zur Definition von Testfällen, um die Zuverlässigkeit von Algorithmen nachzuweisen.