

❑ Aufgabenstellung:

- Einführung in UML
- Übersicht über verfügbare Diagrammtypen
- Tiefere Betrachtung von Use-Case- und Sequenz-Diagrammen

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 24.04.2018

❑ Übersicht:

- Das Klassendiagramm bietet von allen Diagrammen der Unified Modelling Language (UML) die am meisten detaillierte Sicht auf die Architektur objektorientierter Zusammenhänge.

❑ Aufgabenstellung:

- Darstellen der wichtigsten Eigenschaften, syntaktischen Elemente und der vermittelten Sicht auf ein Model anhand geeigneter Beispiele.

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 24.04.2018

❑ Aufgabenstellung:

- Einführung in objektorientierte Entwicklung
- Unterschiede zu, Vorteile und Nachteile ggü. anderen Paradigmen?
- Wie verhalten sich Instanzen der Objekte zur Laufzeit?
 - Vererbung
 - Polymorphie

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 24.04.2018

□ Übersicht:

- Mit Enterprise Architect gibt SparxSystems den SW-Architekten ein mächtiges Werkzeug für die Modellierung komplexer Softwaresysteme an die Hand

□ Aufgabenstellung:

- Demonstration: Einrichten von EA und Nutzung der im Praktikum verwendeten Modelle
- Demonstration: Erstellung von Use-Case-, Sequenz- und Klassendiagrammen

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 24.04.2018

❑ Überblick:

- Kompilieren wird häufig als Überbegriff für eine Reihe nacheinander ablaufender Prozesse und Aktionen verwendet aus denen am Ende ein ausführbares Programm entsteht.

❑ Aufgabenstellung:

- Darstellen der Schritte vom Code zum Programm
 - Präprozessor
 - Compiler
 - Linker
 - ARM-Crosscompiler (Linaro Toolchain)

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 08.05.2018

❑ Aufgabenstellung:

- Wie funktioniert eigentlich ein Buildsystem?
 - Libraries und Anwendungen
 - Compiler-Einstellungen
- Beispiele anhand von CMake
 - Testprojekt mit (statischer) Library und Anwendung

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 08.05.2018

❑ Aufgabenstellung:

- Vorstellen der Grundlagen und Guidelines zum Produzieren von gutem, lesbaren C++ Code.
- Einführung: was ist Doxygen?
- Vorstellen der Doxygen-Syntax
- Beispiele für Verwendung und daraus erzeugter Dokumentation

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 08.05.2018

❑ Übersicht:

- Mit Eclipse steht den SW Entwicklern ein mächtiges Werkzeug für u.a. Coding, Debugging und Compiling an die Hand

❑ Aufgabenstellung:

- Darstellen der Benutzung von Eclipse für SW-Entwicklung mit C++
 - Anlegen eines Projektes
 - Schreiben von Code
 - Übersetzen des Programms
 - Debugging
 - Interpretieren von Fehlermeldungen

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 15.05.2018

❑ Übersicht:

- Collective Code Ownership und Versionierung sind nicht mehr wegzudenkende Methoden bei Softwareentwicklungen mit mehreren Programmierern. Sie erlauben die gleichzeitige Bearbeitung von Code von mehreren Entwicklern sowie das zurückgehen zu vorangegangenen Ständen.

❑ Aufgabenstellung:

- Darstellen der Grundzüge und Funktion der Versionsverwaltung (Clone, Commit, Branching, Merge, Konflikte)
- Vorstellen des Tools Tortoise Git sowie Benutzung
- Vorstellen der Eclipse-Integration sowie Benutzung

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 15.05.2018

❑ Übersicht:

- Framework für C++ Tests auf verschiedenen Plattformen (Linux, Mac OS X, Windows, ...) von Google, basierend auf der xUnit Architektur.
 - Wird u.a. eingesetzt im Chromium Projekt und dem LLVM Compiler

❑ Aufgabenstellung:

- Einführung: was sind Unit Tests?
- Vorstellung des googletest Frameworks
- Beispiele für die Verwendung und Testdurchführung

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 15.05.2018