

Übersicht Vortragsthemen

- ☐ UML Einführung / Use-Case- & Sequenz-Diagramme
- ☐ UML / Klassendiagramme
- ☐ Einführung in Enterprise Architect
- ☐ Objektorientierte Entwicklung
- ☐ Grundlagen C++
- ☐ Coding Style Guidelines (C++) und Doxygen
- ☐ Entwurfsmuster
- ☐ Versions-Management: Git: Prinzip & Clients
- ☐ Betriebssystem: Grundkenntnisse Linux
- ☐ Entwicklungsumgebung: Cmake
- ☐ Entwicklungsumgebung: Visual Studio
- ☐ Graphische Ausgabe mit SDL
- ☐ Unit Test Framework „googletest“
- ☐ Integration- und Abnahmetest

❑ Aufgabenstellung:

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

Bearbeiter 1: Dennis Schuckmann

Bearbeiter 2: Michael Mahl

Präsentationsdatum: 30.04.2020

❑ 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: Prakashan Palendran

Bearbeiter 2:

Präsentationsdatum: 30.04.2020

❑ 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: 30.04.2020

❑ 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: 07.05.2020

❑ Übersicht:

- Überblick über die Grundlagen der Programmiersprache C++

❑ Aufgabenstellung:

- Einführung: Was macht die Sprache C++ aus?
- Überblick über die wichtigsten Konstrukte in C++
- Besonderer Schwerpunkt auf Speicher bzw. Speicherzugriff

Bearbeiter 1: Matthias Stammer

Bearbeiter 2:

Präsentationsdatum: 07.05.2020

❑ 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: 07.05.2020

❑ Übersicht:

- **Entwurfsmuster** (englisch design patterns) sind bewährte Lösungsschablonen für wiederkehrende Entwurfsprobleme sowohl in der Architektur als auch in der Softwarearchitektur und -entwicklung. Sie stellen damit eine wiederverwendbare Vorlage zur Problemlösung dar, die in einem bestimmten Zusammenhang einsetzbar ist. In den letzten Jahren hat der Ansatz der Entwurfsmuster auch zunehmendes Interesse im Bereich der Mensch-Computer-Interaktion gefunden.

❑ Aufgabenstellung:

- Einführung: Was sind Entwurfsmuster?
- Vorstellung der wichtigsten Entwurfsmuster
- Herausarbeitung der wesentlichen Unterschiede der einzelnen Muster

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 14.05.2020

❑ 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 Visual-Studio-Integration sowie Benutzung

Bearbeiter 1: Stephan Rumswinkel

Bearbeiter 2:

Präsentationsdatum: 14.05.2020

□ Übersicht:

- Unterschied Linux/ Windows
- Kommandozeile
- Zugriff via SSH & SFTP/ SCP
- X-Forwarding
- Paketverwaltung
- Umgebungsvariablen
- udev für USB-Geräte
- Zielsystem im Praktikum: Ubuntu 18.04 LTS

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 14.05.2020

□ Übersicht:

- Wie funktioniert eigentlich ein Buildsystem?
 - Libraries und Anwendungen
 - Compiler-Einstellungen
- Beispiele anhand von CMake
 - x86_64-Testprojekt mit (statischer) Library und Anwendung
 - Generierung von Visual-Studio-Projektdaten
- Vcpkg
 - Pakete nachinstallieren
 - Pakete in CMake verwenden (Beispiel: googletest, boost)

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 28.05.2020

□ Übersicht:

- Benutzerinterface, z.B.
 - Solutionexplorer
 - Hilfsmechanismen (z.B. Refactoring)
 - Git-Integration (ggf. mit Vortrag zu Git koordinieren)
- CMake-generierte Projektdatei importieren
- Debugging
 - Compiler-Fehlermeldungen
 - Starten eines Programms mit Debugger
 - Breakpoints und Watch
 - Callstack

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 28.05.2020

❑ Übersicht:

- SDL steht für "Simple DirectMedia Layer" und ist eine portable Open Source-Multimedia-Bibliothek. "Portabel" bedeutet, dass Anwendungen, die (ausschließlich) auf SDL setzen, ohne Probleme auf verschiedenen Plattformen wie Windows, Linux oder Mac OS kompiliert und ausgeführt werden können. Die Bibliothek ist in C geschrieben, kann aber natürlich auch unter C++ genutzt werden

❑ Aufgabenstellung:

- Einführung: was sind SDL und wofür wird es genutzt?
- Vorstellung von einem Anwendungsbeispiel
- Welche Vorteile bietet SDL

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 28.05.2020

❑ 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: 18.06.2020

❑ Übersicht:

- Integration- und Abnahmetests sind eine wichtige Komponente während dem Software- Entwurfsprozess.

❑ Aufgabenstellung:

- Einführung: was sind Integrations- und Abnahmetests?
- Warum sind diese Tests so wichtig?
- Beispiele für die Verwendung und Testdurchführung

Bearbeiter 1:

Bearbeiter 2:

Präsentationsdatum: 18.06.2020