

Programmmentwurf

TINF19B5

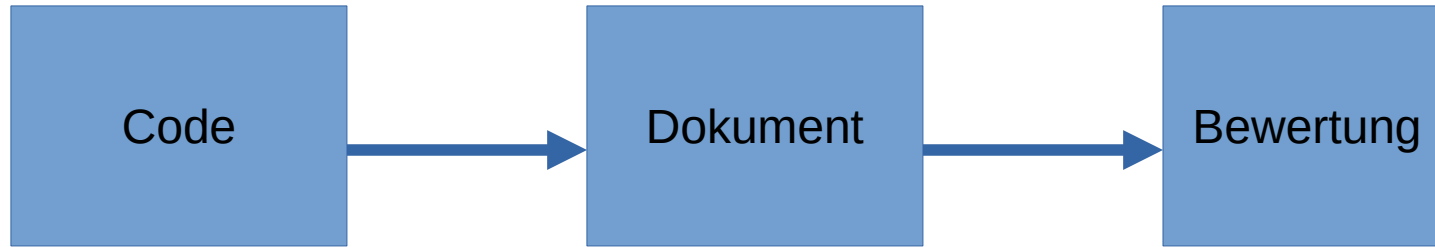
Umfang und Bewertung

Programmentwurf

1.1.8 Programmentwurf (PE)

Ein Programmentwurf umfasst die Bearbeitung einer Aufgabenstellung mit der Auswahl geeigneter Methoden, der Formulierung der verwendeten Algorithmen in einer Programmiersprache, das Testen und Überprüfen der Ergebnisse auf Richtigkeit und die Programmdokumentation.

https://www.dhbw.de/fileadmin/user_upload/Dokumente/Amtliche_Bekanntmachungen/2019/14_2019_StuPrO_Technik_inklusive_Dritter_Aenderungssatzung.pdf



- Es wird eine Applikation entwickelt und ein dazugehöriges Dokument geschrieben
- Dokument baut auf dem Source Code auf und wird am Ende bewertet
- ≥ 2.000 Zeilen Code
- Java, C#
- ≥ 20 Klassen
- keine GUI, nur textbasierte Ausgabe

Source Code

- Versionskontrolle (git)
- vollständiges Repository abgeben (hochladen)
- vollständiger Source Code
 - kompilierbar
 - testbar
 - ausführbar
- “Works on any (reasonable) machine.”
 - mein PC: aktuelles Manjaro-Linux
- 3rd-Party-Libraries erlaubt
- Frameworks (z.B. Spring) sind nicht erlaubt

Source Code (Fortsetzung)

- Build Tools (Maven, Gradle, Make, ...) sind erlaubt
- Docker zur Auslieferung ist möglich, wenn der Container unter Linux läuft
- die Applikation sollte mit einem Klick / einem Script vollständig starten

Schriftliche Dokumentation

- Die Struktur der Vorlage muss benutzt werden (das Dokument kann aber mit jedem beliebigen Programm erzeugt werden)
- PDF-Dokument
 - keine Mindest- oder Maximalseitenzahl
 - kein akademischer Text
- keine Benutzerdokumentation, sondern technische Dokumentation
- enthält Ergebnisse der Analysen und Begründungen der Entscheidungen
- Code-Struktur als UML-Diagramme notiert
 - Detailgrad an die Aussagen anpassen

Themenvorschläge

- Kalender
- Eventmanager
- Textadventure
- Musikplayer
- Email-Client
- Chat (P2P, Server/Client)
- Filesharing (P2P, Server/Client)
- Backup-Manager
- **Eigene Ideen einreichen!**
- First come, first serve

Gruppenarbeit

- Gruppenarbeit ist prinzipiell erlaubt
- max. 2er Gruppen
 - die Einzelleistung muss sichtbar sein
- alle Commits müssen im Repository vorhanden sein (um nachvollziehen zu können, wer was umgesetzt hat)
 - es muss also jeder seine Teile mit seinem Account “comitten”
- die Gruppen werden zusammen bewertet

Gruppenarbeit (Fortsetzung)

- die schriftliche Dokumentation wird zusammen geschrieben
- die Aufgaben können aufgeteilt werden, aber jeder muss mindestens einen Teil der Aufgaben lösen
- bei sichtbaren Unstimmigkeiten werden die Studierenden einzeln mündlich befragt und es kann dann eine voneinander abweichende Bewertung geben

Abgabe

- vollständige Abgabe muss bis einschließlich 31.5. statt finden
 - Abgabe heißt: Zugang zum Git Repository, das den Code und das Dokument enthält
 - der Zugang kann natürlich direkt zu Beginn geschickt werden

Einreichung der Themen

- Themen bis einschließlich 14.11. per Mail an maurice.dhbw@mamem.de
- Stichpunktartig die Technologie und die Eigenschaften auflisten
 - z.B.
 - HTTP Server zur ToDo-Listen-Verwaltung
 - Java, Gradle
 - REST-API mit XYZ
 - CRUD Operationen für Einträge
 - ...