

Handreichung Softwareprojekt 2019

Anforderungen & Bewertungskriterien

- Es wird euch ein Uni-internes gitlab Repository bereitgestellt, in welchem die von euch erstellte Software, Dokumentationen, Tests, etc. gespeichert werden müssen.

Pflichtenheft + Projektzeitplan + Milestones

- Pflichtenheft: Bitte seht euch dazu das von uns für euch im Repository bereitgestellte Beispieldokument *Pflichtenheft_Beiispiel.pdf* an und greift auf euer Vorwissen aus der Veranstaltung *Software Engineering* zurück.
- Projektzeitplan: Als Anhang an das Pflichtenheft oder als gesondertes Dokument ist von euch ein Projektzeitplan abzugeben, welcher für jede der 13 Projektwochen angibt, was ihr in dieser Woche erledigen wollt bzw. was bis dahin insgesamt erledigt sein soll.
- Milestones: Ihr seid angehalten, drei Milestones zu spezifizieren. Bei diesen handelt es sich um Spezifikationen, was bis zur jeweiligen Deadline erledigt sein soll. Die Deadlines sind vorgegeben, ihr müsst den Milestones lediglich einen aussagekräftigen und verhältnismäßigen Inhalt zuweisen.
 - Milestone 1: Deadline ist **Sonntag, der 19. Mai 2019.**
 - Milestone 2: Deadline ist **Sonntag, der 09. Juni 2019.**
 - Milestone 3: Deadline ist **Sonntag, der 07. Juli 2019.**
- Die Deadline für das erste "Abgabepaket" (Pflichtenheft + Projektzeitplan + Milestones) ist **spätestens zwei Wochen nach eurem initialen Meeting in KW 17.**
- Ihr dürft diese Dokumente auf **englisch** verfassen, wenn ihr wollt - könnt es aber auch gerne auf **deutsch** machen. Diesbezüglich gibt es keine Vorschrift.
- Abgabe erfolgt im Repository im dafür vorgesehenen Ordner.

Design-Spezifikation

- Bitte seht euch dazu das von uns für euch im Repository bereitgestellte Beispieldokument *Designspezifikation_Beiispiel.pdf* an und greift auf euer Vorwissen aus der Veranstaltung *Software Engineering* zurück.
- Die Deadline für die Designspezifikation ist **spätestens Sonntag, der 19. Mai 2019.**
- Ihr dürft dieses Dokument auf **englisch** verfassen, wenn ihr wollt - könnt es aber auch gerne auf **deutsch** machen. Diesbezüglich gibt es keine Vorschrift.
- Abgabe erfolgt im Repository im dafür vorgesehenen Ordner.

2-Wochen-Berichte

- In den ersten 12 Wochen des Projektes müsst ihr alle 14 Tage einen Kurzbericht in euer Repository hochladen. Dieser Bericht sollte folgendes enthalten und eine Länge von einer Seite nicht überschreiten:
 - Was ihr in den letzten 14 Tagen generell erledigt habt.
 - Was jedes einzelne Gruppenmitglied davon erledigt hat.
 - Welche wichtigen Entscheidungen ihr in den letzten 14 Tagen getroffen habt (und warum!).
- Abgabe erfolgt im Repository im dafür vorgesehenen Ordner.

Code + Tests + Docs

- Selbstverständlich ist das Herzstück des Projektes der von euch generierte Code.
- Neben dem reinen Programmieren solltet ihr aber auch darauf achten, eure Software ausgiebig zu testen und diesen Prozess zu dokumentieren. Wir empfehlen, auf jeden Fall unit tests zu schreiben, welche dann auch im Repository verfügbar sind, damit auch wir diese laufen lassen können.
- Dokumentation: Neben Code und Tests sollt ihr am Ende auch eine Dokumentation produziert haben. Dies umfasst einerseits eine Anwender-Doku wie z.B. ein Handbuch und/oder ein **getting started** Dokument - und andererseits eine API Dokumentation wie man sie beispielsweise von Oracle für Java kennt. Dazu könnt ihr für die Java-Projekte den **Javadoc** Standard verwenden und für die TEAMWISE-Projekte den **jsdoc** Standard.

Zwischenevaluation

- In der Mitte des Semesters wird es Zwischenevaluationsgespräche geben. Diese dienen einerseits dazu, dass wir, wenn nötig, eingreifen und unterstützen, können - und andererseits dazu, eine erste Einschätzung eurer Arbeit vorzunehmen und euch entsprechendes Feedback zu geben.

Meeting minutes

- Während den Betreuer-Meetings schreibt mindestens einer von euch mit. Das daraus entstehende, ggf. gekürzte, Protokoll von maximal einer Seite Länge ladet ihr ebenfalls ins Repository hoch. Dies sollte spätestens drei Tage nach dem Meeting erfolgen.
- Abgabe erfolgt im Repository im dafür vorgesehenen Ordner.

Organisation innerhalb der Gruppe

- Ihr solltet euch innerhalb der Gruppe selbst organisieren. Welche Rollen und Verantwortlichkeiten ihr dazu vergebt und ob ihr diese fest lasst oder rotiert - bleibt im wesentlichen euch überlassen. Wichtig ist nur,
 - o dass für jedes der Betreuer-Meetings ein Protokollant festgelegt ist, und
 - o dass es stets ein Gruppenmitglied gibt, welches als Ansprechpartner für die Betreuer fungiert. Diese Rolle sollte immer mindestens 4 Wochen lang von einer Person eingenommen werden - und uns jeweils zeitnah mitgeteilt werden.
- Ihr solltet mindestens einmal pro Woche ein gruppeninternes Meeting haben, indem ihr euch besprechen könnt.
 - o Auch bei diesen Meetings muss einer aus der Gruppe Protokoll führen und ein ggf. gekürztes Protokoll von maximal einer Seite Länge ins Repository in den dafür vorgesehenen Ordner hochladen.
 - o Als Projektgruppe im Software-Projekt habt ihr die Möglichkeit, bei der universitären LEhrraumvergabe einen Seminarraum für einen fixen wöchentlichen Termin zu reservieren. Dazu könnt ihr eine Mail an Frau Laub unter **Lehrraumvergabe@uni-konstanz.de** senden. Weiter Infos gibt's hier: <https://www.uni-konstanz.de/facility-management/gebaeudeservices/raumverwaltung-gaestehaeuser-schluesselverwaltung-veranstaltungsplanung/lehrraumplanung/>.

Produktivitäts-Metriken

- Ihr bekommt ein von der Uni gehostetes gitlab-Repository. In diesem werdet ihr euer Projekt entwickeln und verschiedene, oben genannte Dokumente etc. hinterlegen. Drei wichtige Informationen dazu:
 - o Gitlab besitzt einen integrierten Issue-Tracker (und ein issue board), welcher dazu verwendet werden kann, issues anzulegen und diese wiederum ggf. mit einer Deadline, Gruppenmitgliedern zuzuweisen. Ihr sollt diese Funktion nutzen, damit wir stets euren Fortschritt und eure Arbeitsverteilung im Auge behalten können.

- Außerdem gibt es die Möglichkeit in gitlab Milestones anzulegen. Diese Funktionalität sollt ihr verwenden, um eure Milestones zu hinterlegen. Weiterhin gibt es die Möglichkeit, einem Milestone issues zuzuordnen, um den Fortschritt im Hinblick auf die einzelnen Milestones im Auge zu behalten.
- Wie bereits in der Auftaktveranstaltung gesagt wurde, muss jedes Gruppenmitglied mitprogrammieren, ungeachtet der internen Gruppenorganisation. Um dies wiederum transparent zu halten, ist es wichtig, dass jeder **Commits** im Repository macht. Wir werden zur Zwischen- und Endevaluation die commit-Statistiken (sowohl Anzahl der commits als auch Code-Umfang der einzelnen Commits) im Repository prüfen. Solltet ihr also Techniken verwenden, bei denen ihr in Paaren oder größeren Gruppen gemeinsame coding sessions habt - so solltet ihr auch darauf achten, diese commits dann gleichmäßig auf die einzelnen Gruppenmitglieder aufzuteilen, um Missverständnisse zu vermeiden.

Abschluss-Präsentation

- Am **Dienstag, den 23.07.2019** werden die Abschlusspräsentationen stattfinden. Diese umfassen eine Projektpräsentation von etwa 12 Minuten pro Gruppe. Dabei soll etwa die Hälfte der Zeit auf eine Folienpräsentation entfallen, während die zweite Hälfte der Zeit eine Software-Demo umfasst.

Lizenzen

- Ein sehr wichtiges, und doch oft unterschätztes, Thema sind Software-Lizenzen. Da auch die von euch produzierte Software unter einer Open Source Lizenz entwickelt wird, ist es enorm wichtig, dass ihr bereits während dem Entwicklungsprozess Buch darüber führt, welche externen Komponenten ihr für eure Software verwendet und unter welchen Lizenzen diese angeboten werden. Dies umfasst **unter Anderem auch**
 - externe libraries und frameworks
 - verwendete Bilder und Icons
 - verwendete Schriftarten etc. Dazu sollt ihr im Repository ebenfalls ein Textdokument anlegen, und darin all diese Komponenten und ihre zugehörigen Lizenzen auflisten.

Weiteres

Tutoren

Erste Anlaufstelle bei jeglichen Problemen bzw. Fragen zu Tools (ibs. Eclipse, Visual Studio Code, git, ...) oder Entwicklungs-Workflows sind unsere Tutoren Benjamin Möck und Wilhelm Kerle. Ihr könnt die beiden jederzeit per Mail anschreiben und einen individuellen Betreuungs-Termin vereinbaren. Achtet bitte darauf, in diesem Fall die Mail immer an beide Tutoren zu adressieren, damit die beiden sich absprechen und in geeigneter Weise reagieren können.

- Benjamin Möck: **benjamin.moeck@uni-konstanz.de**
- Wilhelm Kerle: **wilhelm.kerle@uni-konstanz.de**

Ressourcen

Neben den Ressourcen, welche wir für euch im Repository hinterlegt haben, nennen wir hier noch einige weitere.

- Gruppen 1, 2.1, 2.2, 3
 - www.vanted.org : offizielle Vanted-Webseite
- Gruppen 4, 5

- www.cesiumjs.org : offizielle Cesium-Webseite
- <https://cesium.com/docs/tutorials/getting-started/> : Cesium getting started tutorials/getting-started/
- <https://cesium.com/docs/tutorials/cesium-workshop/> : Cesium workshop tutorial
- <https://www.w3schools.com/w3css/> : w3css tutorial / API

Termine und Fristen - kurz & prägnant

Pflichtenheft + Projektzeitplan + Milestones: Spätestens 2 Wochen nach dem initialen Meeting ins Repository unter `Abgaben/Dokumente`.

Designspezifikation: Spätestens am Sonntag, den 19.05.2019 ins Repository unter `Abgaben/Dokumente`.

Milestone 1 | 2 | 3: 19.05.2019 | 09.06.2019 | 07.07.2019.

2-Wochen-Berichte: 05.05.2019 | 19.05.2019 | 02.06.2019 | 16.06.2019 | 30.06.2019 | 14.07.2019 ins Repository unter `Abgaben/ZweiWochenBerichte`.

Betreuer-Meeting-Protokolle: Jeweils spätestens 3 Tage nach dem Betreuer-Meeting ins Repository unter `Protokolle/BetreuerTreffen`.

Gruppeninterne Meeting-Protokolle: Jeweils spätestens 3 Tage nach eurem **wöchentlichen** gruppeninternen Meeting ins Repository unter `Protokolle/InterneTreffen`.