

Software Engineering II

Übergabeprotokoll

Gruppe C

Bauhaus-Universität Weimar

21. Januar 2024

Reflexion

Themen, die uns geholfen haben

Requirements

Die Vorlesungen „Requirements Engineering“ und „Representing the requirements“ haben uns in unserem Software Development Prozess sehr geholfen.

Für den Start war insbesondere der Abschnitt „do requirements come from“ hilfreich. Er hat uns dabei geholfen unsere ersten Ideen zu unserem Projekt zu organisieren und diese Ideen in Requirements umzuformulieren. Desweiteren hatten wir durch die Vorlesungen einen guten Überblick über alle Bereiche unseres Projekts, über die wir nachdenken und zu denen wir Requirements formulieren mussten.

Es hat uns auch geholfen, dass über die verschiedenen Arten Requirements zu notieren gesprochen wurde. Zu jeder Art wurde gut erklärt, wie die Notation normalerweise aussieht und was aber auch Vor- und Nachteile der Notationsart sind. So hatten wir einen guten Überblick und konnten uns letztlich für die Art, mit der wir am besten klarkamen, entscheiden. Besonders durch das Besprechen der Probleme der Notationsarten konnten wir unsere Requirements im Anschluss nochmal überarbeiten.

Code Quality

Die Vorlesung „Code Quality“ war sehr hilfreich für uns.

Es passiert doch sehr häufig, dass der erste geschriebene Code unübersichtlich ist und eine schlechte Qualität hat, und man vergisst diese Fehler auszubessern. Uns ist aber auch aufgefallen wie störend diese Fehler sind, wenn man den Code überarbeiten muss.

Es hat uns geholfen, dass wir eine Übersicht über diese Fehler bekommen haben und dass auch offensichtlich erscheinende Fehler explizit angesprochen wurden. So können wir sie hoffentlich leichter erkennen und vermeiden

Themen die uns gefehlt haben

Software Patterns

Im vergangenen Semester (SE1) wurden sehr marginal auch Softwarepatterns angerissen und diese erscheinen uns daher im Hinblick auf die Belege als sehr hilfreiches Werkzeug, um den Code besser und grundlegend durchdachter aufbauen zu können. Deshalb hätten wir den Wunsch gehabt, dass insbesondere im Hinblick auf die Softwareentwicklung im Team und auf industrieller Basis, es sehr hilfreich sein könnte, Code nach bekannten und funktionierenden Strukturen bauen zu können und auf althergebrachte Werkzeuge zurück greifen zu können. Ein solches Werkzeug stellen die Software Patterns dar, diese wurden dafür geschaffen, grundlegende Programmstrukturen für häufig auftretende Probleme modellieren zu können.

Ausführungen des Whitebox Testing

Ein Punkt der uns gefehlt hat waren genauere Ausführungen bzgl. des Whitebox Testing.

In der Vorlesung wurde Whitbox Testing sehr theoretisch besprochen und die Prinzipien hinter Whitebox Testing hatten wir soweit auch verstanden. Als wir dann selber Whitebox Tests implementieren sollten haben wir uns jedoch sehr schwer getan.

Wir wussten nicht, wie man in der Praxis und vorallem mit Whitebox Testing vorgeht. Das Video, was wir zum Belegaufgabenblatt bekommen haben war hilfreich, jedoch nur für einfachere, kürzere Funktionen. Über das Vorgehen bei längeren, komplizierten Funktionen waren wir uns immer noch nicht bewusst.

Wir hätten uns genauere Erklärungen zur praktischen Anwendung von Whitebox Testing gewünscht, aber in diesem Zusammenhang auch zum Thema Refactoring. Wir haben unseren Code refactored aber wie uns im Nachhinein erklärt wurde, nicht besonders gut und zielführend. Tipps oder Techniken, wie den Code für das Whitebox Testing vorbereitet und die Qualität des Codes verbessert, hätten uns sehr geholfen.