

Persönlicher Bericht von Marcel Hess

GENERELLER EINDRUCK

Die Bachelorarbeit in diesem Semester hat mir allgemein sehr viel Spass bereitet. Es war sehr spannend die verschiedenen, theoretisch erlernten Aspekte des gesamten Studiums in diesem Projekt praktisch einfließen zu lassen und entsprechend erste Erfahrungen in einem grösseren Projekt zu erlangen.

Anders als bei der SA, konnten wir uns in der BA primär wirklich oft nur aufs Entwickeln an sich konzentrieren, ohne parallel Infrastrukturprobleme (wie beispielsweise den CI Prozess) im Nacken zu haben. Durch das praktische Anwenden konnte ich auch wirklich mal erleben, wie es in einem grösseren Projekt ablaufen kann, wenn man von verschiedenen Automatisierungsprozessen (wie eben namentlich das lauffähige CI) profitieren kann. Entsprechend möchte ich auch in Zukunft diesen Komfort geniessen und ich habe nun eine ganz andere Einstellung zum initialen Aufsetzen von automatisierten CI Prozessen erlangt.

VERBESSERUNGEN ZUM PROZESS

Während des ganzen Projekts hatten wir etwas mit grundlegenden Refactorings zu kämpfen. Ich denke, diesen Aspekt hätte man vermeiden können, in dem wir am Anfang nochmals die ganze Architektur unserer Serverkomponenten - in Hinsicht auf die Symboltabellenerweiterung - hinterfragt hätten.

Im Client haben wir relativ baldig bemerkt, dass das stetige Anpassen der Komponenten sehr ineffizient ist, und haben kurzerhand eigene Komponenten von Grund auf erstellt. Beim Server haben wir dies leider etwas versäumt. Daher haben wir viel Zeit für fortlaufende Umstrukturierungen aufgewendet. Ausserdem war das Isolieren und Umhängen der Abhängigkeiten teils sehr mühsam, da wir nicht gegen die Interfaces programmiert hatten. Diese Zeit hätte man gewinnen können, und für noch besser ausgebaute Features und eine noch bessere Testabdeckung einsetzen können. Die Testabdeckung hat unter diesen fortlaufenden Refactorings auch etwas gelitten. Zwar haben wir bei der Kernlogik weiterhin eine gute Testabdeckung erreicht, aber durch Methoden, die Daten von der einen Komponenten zur nächsten durchreichen, haben wir teils keine 100% Testabdeckung erzielt, was etwas schade ist. Hätten wir für diesen Aspekt noch etwas mehr Zeit aufwenden können, wären die Testabdeckungszahlen noch viel besser geworden.

TEAM

Die Zusammenarbeit mit Thomas Kistler war bereits während der SA äusserst angenehm und auch in der BA wurde diese Arbeitsweise fortgesetzt. Teils haben wir die sehr komplexen Codestellen in Pair-Programming implementiert - beispielsweise das Grundgerüst der Symboltabelle. Wir haben aus Effizienzgründen aber auch gewisse Arbeiten aufgeteilt, wie beispielsweise Refactoring- und Ausbauarbeiten von gewissen Komponenten.

LERNEFFEKT

Einer meiner grössten Lernschlussfolgerungen, welche ich aus der BA für meine Zukunft mitnehme, ist das Programmieren gegen definierte Interfaces. Wir hatten das zwar schon öfter *gehört in Vorlesungen*, aber der effektive Nutzen hat sich mir bisher nie so richtig aufgezeigt. Inzwischen weiss ich die Isolation durch Interfaces aber sehr zu schätzen. Ich denke, das war eine klassische Thematik, bei welcher man *erst mal auf die Nase fallen musste*, um den Hintergedanken von Interfaces wirklich zu verinnerlichen.

BEURTEILUNG DES ENDPRODUKTS

Es freut mich sehr, dass unser Plugin veröffentlicht wurde und andere Entwickler von unserer Arbeit profitieren können. Es ist ein gutes Gefühl, dass wir keine *theoretische Arbeit für den Ablagestapel* produziert haben. Etwas schade ist dennoch, dass wir keinen Mono-Support für den Server erzielt haben, und daher keinen Mergerequest für den Dafny-Teil absetzen konnten. Ich denke die Mono-Vernachlässigung war aber eine richtige Entscheidung, sonst hätten wir ähnlich wie bei der SA wieder ein stetiges *Infrastrukturproblem* im Nacken gehabt.