

I: Q1

B: Also wir machen das CR über ein Tool "*Gerri*", das ist auch verknüpft mit einem anderen Tool "*Crucible*", welches gleich bestimmte Fehler hervorhebt. Es gibt Regeln, die man auch selber definieren kann bzw. Standard Regeln, wie aus JAVA beispielsweise, die da geprüft werden und als Fehler hervorgehoben werden. tool ermöglicht halt direkt die Änderungen zu sehen, die die Kollegen mit dem Commit gemacht haben. Dann kann man besser sehen, was sich geändert hat, und direkt die Kommentare eintragen und dann an den entsprechenden Code stellen, was geändert werden muss. Dann wird am Ende noch eine Bewertung vergeben, ob der Code gemerkt wurde oder, ob was zu tun ist. Dies ist ein Sicherheitstool, so dass man nicht einfach irgendwas pfuschen kann, sondern es muss jemand vorher sein Ok gegeben haben.

I: Q2

B: Wir haben verschiedene Entwicklungsrichtlinien, die zu Beginn des Projekts von unseren Architekten gesetzt wurden. Das sind so Sachen wie beispielsweise: sind die Methoden vernünftig dokumentiert, die Parameter die übergeben werden etc. Wir verwenden die Englische Sprache , d.h. die Sachen müssen vernünftig in Englischer Sprache benannt sein, müssen verständlich sein, man muss aus der Methode z.B. herauslesen können, was diese macht. Ansonsten ist wichtig, dass es natürlich richtig programmiert wurde, dass genügend Tests für die neue Funktionalität gemacht wurden etc.

I: Q3

B: Es muss natürlich auch dem aktuellen Java entsprechen. Wir entwickeln ja mit unserem Projekt Java und es muss natürlich auch gut verständlich und übersichtlich sein. Nicht so viele Schleifen ineinander , so dass man nicht so gut versteht was passiert. Jemand der nichts entwickelt hat, muss verstehen was die Funktion macht.

I: Q4

B: Es ist sehr wichtig, um die Qualität zu halten - das macht ein Qualitätsmanager bei uns. Dadurch dass jemand zweites nochmal drüber guckt, vor allem ein erfahrener Kollege, kann man natürlich Fehler viel schneller erkennen oder auch den Code viel x machen, durch diesen zweiten Blick, sag ich mal, auf den Code - Kann man ja gucken ist er verständlich? Ich als zweite Person, verstehe das, was der andere entwickelt hat und dann kann man die Qualität auch viel höher halten.

I: Also meinst du jetzt, dass wird einfach durch die Übungen oder durch die Reviews bzw. Änderungswünsche?

B: Also das eine ist, der Code wird natürlich besser, dadurch dass ein zweiter nochmal drauf guckt und anderen seine Meinung dazu gibt und andererseits dadurch, dass das Review/Feedback an den Kollegen der entwickelt bringt, lernt er natürlich dadurch aus dazu und vermeidet den Fehler in der Zukunft - so in die Richtung.

I: Q5

B: Das sind sehr unterschiedliche Kollegen, also das sind teilweise Neueinsteiger, die noch nicht soviel Erfahrung haben, andererseits sind das erfahrene Kollegen, u.a. Architekten, die Reviews machen. Also wir machen auch Peer- Reviews , d.h. wir Entwickler Review uns gegenseitig. es ist nicht immer der Architekt der darauf guckt. Es hängt davon ab, wie schwerwiegend die Änderungen sind , ob die größere Auswirkungen hat., wenn es starke Auswirkungen hat, dann macht es der Architekt, um halt auch wirklich sicher zu gehen, ob alles soweit stimmt und ansonsten machen wir halt gegenseitige Reviews. Die Entwickler untereinander. Also wir haben Kollegen aus Polen dabei, aus Deutschland...teilweise sind die schon einige Jahre bei Capgemini oder auch als Entwickler tätig und wieder andere sind neu dazu gekommen.

I: Wie viele seid ihr?

B: Kommt darauf an, also fragst du jetzt in dem ganzen Projekt oder nur in diesem Teilsystem?

I: Nur in diesem Projekt , im Code Review Prozess, die Reviewers und Entwickler?

B: Ca. 10.

I: Die alle entwickeln und reviewen?

B: Ja

I: Q6

B: Am einfachsten ist es natürlich von den erfahrenen Kollegen was zu Reviewen, denn da hat man nicht so viel zu tun. Bei neueren Kollegen muss man schon genauer darauf achten, dass man denen auch richtige Hinweise gibt, damit sie für die Zukunft dazu lernen.

I: Gibt es bestimmte Kollegen?

B: Nicht unbedingt eine bestimmte, eher so Gruppenweise, sag ich mal, die erfahrenen Kollegen sind einfacher als die neuen Kollegen.

I: Q7

B: nein, eigentlich nicht, ist mir eigentlich gleich.

I: Q8

B: Ich bin ja auch erst hier seit letztem Jahr hier bei Capgemini. Und das ist hier ja auch mein erstes Projekt was ich gerade mache. Als wir mit der Entwicklung angefangen haben, war ja noch nichts weiter da, wir haben ja wirklich bei Null gestartet und ich habe eine Komponente bekommen, die nach bestimmten Regeln arbeiten soll, da hatte ich, ja, auch wirklich viel zu tun und hatte ja auch noch nicht so den neusten Java Standard drauf. Die Kollegen die das gereviewed haben, hatten natürlich viele Verbesserungsvorschläge, um den Code auch performanter zu machen, damit diese Regelabfrage schneller läuft, ja da hatte ich auch mehrfach nachzuarbeiten. Weil dann macht da jemand eine Review darauf, dann kommt es zurück zu dir, dann arbeitest du es einstellst es wieder bereit und dann kommt es gegebenenfalls wieder zurück und das kam halt öfter vor, ja.

I: Und was war besonders schwierig daran?

B: Schwer ist es natürlich immer am Anfang, die Richtlinien einzuhalten, also dann gibt es immer so einen schönen Katalog mit Sachen, die gemacht werden sollen und dann müssen die sich natürlich erstmal darauf einstellen und sich das angewöhnen, dass man diese ganzen Regeln einhält.

I: War es für dich schwierig diese Regeln einzuhalten oder für die Anderen?

B: Am Anfang war es für uns alle schwer, das ging dann recht zügig.

I: Die Richtlinien sind von Anfang an festgelegt?

B: Ja, von Anfang an des Projekts sind sie festgelegt und dann gelten sie das ganze Projekt über.

I: Q9

B: Einerseits lerne ich natürlich, wie ich den Code performanter, von der Qualität her besser machen kann, z.B. leserlicher, verständlicher und andererseits schaffe ich durch den Code -Review auch, den Qualitätsstandard zu halten, der an mich gefordert ist. Also jeder Entwickler muss ja einen gewissen Standard liefern, damit das Projekt auch wirklich einen guten Qualitätsstandard hat und dadurch, dass man halt gegenseitig eine Review darauf macht, trainiert man sich selber und den anderen diese Qualitätsstandards einzuhalten.

I: Q10

B: Also bis jetzt waren wirklich alle Code Reviews produktiv, wenn ich z.B. eine Anmerkung gemacht hab, die nicht korrekt war, oder die vielleicht in einem bestimmten Zusammenhang eine negative Auswirkung gehabt hätte auf den Code, dann hat man sich halt zusammengesetzt, darüber gesprochen und dann habe ich wiederum daraus gelernt, um beim nächsten mal bei den Reviews darauf zu achten. Oder bzw. andersrum, wenn die Anmerkung richtig war, der Kollege aber der Meinung war, die stimmt aber nicht, hat man sich auch zusammengesetzt und darüber gesprochen, hat verschiedene Sichtweisen gesehen und der Kollege hat daraus gelernt.

I: Gab es einen Fall, in dem die Diskussion besonders anstrengend war oder es besonders anstrengend war den anderen zu überzeugen, dass etwas beispielsweise doch gemacht werden sollte?

B: Also ich weiß was du meinst, aber das hatte ich bis jetzt nicht, weil dadurch das wir besonders produktiv arbeiten und man auch immer mit bestimmten Fällen nachweisen kann, warum eine Anmerkung jetzt richtig ist, gab es bis jetzt keine großen Probleme.

I: Also hattest du bis jetzt kein negatives Erlebnis?

B: Nein, bis jetzt nicht.

I: gut - Q11

B: Es geht auf jeden Fall um eine Schleife, wo ein Lieferwert eingefügt werden sollte und hätte diese Schleife einen Lieferwert bekommen, so wie sie normalerweise programmiert wird, hätte dies zu Problemen geführt. Und ich hatte halt nicht die ganze Übersicht, über die ganze Funktionalität, weil es sind ja so einzelne Komponenten, die aufeinander aufbauen und dem entsprechenden habe ich nur die

Änderungen gesehen und ich gesagt, da ist was nicht richtig, wohingegen der Kollege den Blick auf die ganzen Komponenten hatte, da er das ja auch entwickelt hat und dann haben wir uns zusammengesetzt und darüber gesprochen, wie sich das auswirken würde und haben uns dann entschieden es so zu belassen.

I: Wie gehst du eigentlich mit Konflikten um?

B: Wir sind da eigentlich ziemlich locker in der Sprachwahl, sag ich mal. Keiner nimmt es einem übel, wie man etwas formuliert, dass ist ganz nach Bedarf. Es gab bis jetzt auch nicht irgendwie Streit oder sowas, dass sich jemand beleidigt gefühlt hat, durch sein Code Review. Das ist alles ziemlich auf einer sachlichen Ebene.

I: Das war von Anfang an so?

B: Ja

I: (Extra Fragen) Hast du eigentlich von Anfang an, bis jetzt, die selbe Formulierung für deine Meinungen behalten?

B: Am Anfang habe ich natürlich etwas formeller formuliert

I: ...