# Qualitätssicherung

Essentiell für den Erfolg eines Software-Projektes ist die Qualität der jeweilig entwickelten Software. Ausschlaggebend ist jedoch nicht diese zu einem gewissen Zeitpunkt zu erreichen, wie beispielsweise vor der letztendlichen Auslieferung an den jeweiligen Kunden. Sondern dafür die Qualität von Beginn des Entwicklungsprozesses an, während der Entwicklungsphase, bis hin zur Fertigstellung der Software und der darauffolgenden Auslieferung zu erhalten. Anwendung findet hier eine vom ganzen Team genutzte und durchgeplante Qualitätssicherung. Folgende übernimmt das Auffinden von Fehlern und schafft folglich ein grundlegendes Vertrauen in das erarbeitete Produkt.

Vermehrt werden hier in der Softwareentwicklung umfangreiche Softwaretests benutzt, jedoch liegt das Hauptaugenmerk nach Absprache des Teams daran schon keine Fehler in die Software zu bringen. Um dies zu erreichen, werden im gesamten Projekt neben dynamischen und statischen Tests zum Finden von Softwarefehlern und zur Qualitätskontrolle des Produktes, eine ganzheitliche Qualitätssicherung verwirklicht. Dazu werden im Team organisatorische und konstruktive Qualitätssicherungsmaßnahmen eingebunden.

Unter den eingeführten konstruktiven Qualitätsscherungsmaßnahmen wird sich im Projekt eine Verhinderung und auch Minimierung der Fehlereinbringung in der Software versprochen. Hierfür gibt es die Möglichkeiten bereits erprobte Module und Architekturen weiterzuverwenden, die Einbindung eines Frameworks für die Entwicklung oder auch darin die Anforderungen an das Produkt durch Prototypen und Tests dieser zu überprüfen. Das Team hat sich dabei entschlossen sich auf die Nutzung des Spring Frameworks zur Entwicklung zu fokussieren. Dafür sprachen zum einen vorerst qualitätssicherungsunabhängige Aspekte, wie eine vereinfachte Vermittlung zwischen Frontend und dem Backend, Zugriff auf die Datenbank, sowie die vereinfachte Login Erstellung. Jedoch verhindert die Einbindung eines solchen Frameworks von vorn herein schwerwiegenden Fehlern am Anfang der Entwicklung. Da genau die Vorgefertigten und verfügbaren Methoden in das Projekt mit eingebaut werden.

Hingegen sorgen Organisatorische Qualitätssicherungsmaßnahmen für eine Grundlage der gesamten Qualitätssicherung im gesamten Team, da die Ablauf- und Aufbauorganisation eines gemeinsam erarbeiteten Projektes großen Einfluss auf die gelieferte Qualität hat. Aus diesem Grund wird diese zum wichtigsten Punkt der Qualitätssicherung im Projekt, um eine solide Basis für den Erfolg dieses Projektes zu sichern. Erfolgsabhängig für eine erfolgreiche organisatorische Qualitätssicherung stellt hierbei die verwendete Projektmanagementmethode und das genutzte Vorgehensmodell zur Qualitätssicherung. Dabei stand zum einen zur Auswahl eine klassische Software Projektmanagementmethode und das V-Modell als allgemeines Vorgehensmodell zu nutzen. Hierfür wurde im Team die Methode Scrum gewählt, da dieses sich als Rahmen einer ganzheitlichen Qualitätssicherung am besten eignet und vor allem Scrum durch dessen Ablaufs eine bereits einfache Qualitätssicherung besitzt. Zunächst ist Scrum ein einfaches Prozessrahmenwerk zum Projektmanagement, worin erlaubt ist verschiedene Techniken, Methoden und Prozesse zur Entwicklung zu integrieren. Aber trotz dieser hohen Freiheitsgrade besitzt Scrum schon eine Unmenge an im Projekt hilfreichen Qualitätssicherungsmaßnahmen. Genauer gibt es in erster Linie die Ablauf- und Aufbauorganisation des bearbeiteten Projektes vor. Somit entsteht durch die Anwendung des Scrum Ablaufes eine allgemeine und einfache Qualitätssicherung und Qualitätskontrolle. Daher wurde im Team ganzheitlich entschieden die Qualität der Software nur unter der Anwendung der Projektmanagementmethode Scrum zu sichern.

Um die einfache Qualitätssicherung in dieser agilen Projektmanagementmethode zu erläutern, wird ein Blick in den Ablauf und Teile eines Scrum-Projektes geworfen. Darin stellt das gesamte Team schon einmal einen Großteil der Qualität des Projektes dar. Weil durch dieses selbstorganisierte Team die Anforderungen erst realisiert werden können. Somit ist die Zusammenstellung und die Aufteilung einer der kritischen Erfolgsfaktoren für das gesamte Projekt, weshalb des Projekt in Teams für Backend, Frontend und das Mapping spezifisch derer Fähigkeiten eingeteilt wurde. In Hinblick auf die Qualitätssicherung können hier im Team zwei Maßnahmen unterschieden werden. Dass der Produt Owner jederzeit im Laufe der Projektentwicklung voll entscheidungsfähig ist. Und zudem, dass die Teammitglieder neben den Fähigkeiten zur Entwicklung qualitativ hochwertiger Software, auch Kenntnisse in den Methoden der Qualitätssicherung besitzen. Das ist wichtig, da in Scrum eine bestimmte Person bzw. ein dedizierter Tester in Scrum nicht vorhanden ist. Dies wurde über das gesamte Projekt hinweg durchgezogen und schon vor des Entwicklungsstarts dem Team grundlegende Qualitätssicherungsmaßnamen erläutert.

Während der Projektinitiierung übernahm das Team die Aufgabe die Rahmenbedingungen für die weitere Entwicklung des Projektes festzulegen. Dabei wurden im Sinne der Qualitätssicherung Vorgaben hinsichtlich von Programmierrichtlinien oder andern festgelegt, damit eine einfache Wartbarkeit der Software zu gewährleisten war. Im Team ist beispielsweise eine solche Richtlinie, dass keine Software die nicht zu 100% läuft und getestet wurde, in das Teaminterne Git geladen wird. All diese Vorgaben des Teams zusammengefasst bilden die Definition of Done, welches die Kriterien für die fertige Software stellt. Im Team wurde diese Defintion of Done in einer Exel Tabelle einfach gehalten.

Für die Qualitätssicherung stellte auch die Sprintplanung eine sehr große Rolle. Dadurch wurden im gesamten Team die zu implementierenden Anforderungen einem Review unterzogen, um sicherzustellen, dass nur die in der Sprintplanung geplanten und verstanden Anforderungen während des Sprints ausgeführt wurden. Somit wurden klar im gesamten Team die Rollen und die jeweiligen Aufgaben verteilt, um eine qualitativ hochwertige Weiterentwicklung überhaupt zu ermöglichen.

Diese Sprints wurden wöchentlich in einem Meeting teamintern geplant und folglich ausgeführt. Dabei zählten zu den Qualitätssicherungsmaßnahmen während dieses Sprints die zum einen statische Analyse des Codes und Testen des Codes. Sowie die Anwendung des aus Extreme Programming bekannte Pair Programming. Dabei nahm eine Person aus dem Team die Programmierarbeit vor und ein zweiter den geschriebenen Code per Sichtprüfung überprüfte. Somit konnte eine klare Qualitätssicherung während des jeweiligen Sprints erfolgen.

Nach jedem Sprint fand an jedem Donnerstag in der Woche das Teaminterne Teamreview statt. Dabei wurde das im Sprint erarbeitet Produktinkrement vorgestellt und anhand der Anforderungen und der Definition of Done vom jeweiligen Produkt Owner bewertet. Dadurch konnte gewährleistet werden, dass das richtige Produkt entwickelt wird und der nächste Sprint auf einer richtigen Basis stattfindet. Teil eines jeden Sprintreviews ist jedoch auch die weitere Sprintplanung und die Rollenversteilung auf die jeweilig anfallenden Aufgaben gewesen. Wodurch eine Teaminterne und qualitativ hochwertige weiterführende Entwicklung stattfinden konnte.

Um dafür zu sorgen, dass nicht nur die Qualität des Projektes und der Arbeitsweise im Team erhalten und gesichert wurde, hat das Team sich einer Sprint-Retrospektive unterzogen. Dabei wurde zunächst wie in jedem Sprint-Review das entwickelte Produktinkrement und die erreichten Fortschritte gezeigt, jedoch diese Punkte und vor allem auch die Abläufe der im Sprint vollzogenen Entwicklung beobachtet und analysiert. Dadurch konnte im ganzen Team die Arbeitsweise verbessert werden und dadurch die allgemeine Projektqualität erhöht werden. Diese wurde aber lediglich klein gehalten und gemeinsam darüber gesprochen, was verbesserbar wäre. Und nicht genaue Ablaufs Analysen mit Unmengen an ausgearbeiteten Daten erstellt.

Um zuletzt vor der Auslieferung die endgültige Funktion und somit auch die endgültige Vollständige der Software zu gewährleisten, wurde die mit Scrum durchgeführte Qualitätsicherung noch einmal um verschiedene statische und dynamische Softwaretest erweitert. Damit das fertige Projekt voll und ganz funktionsfertig ist.Durch Anwendung dieses Scrum Ablaufes war es möglich die Qualität der entwickelten Software von Beginn der Projektinitiierung bis hin zu der Auslieferung an den Kunden zu erhalten. Da diese Projektmanagementmethode eine schon sehr genaue Qualitätssicherung hat, wurden keine weiteren qualitätssichernden Maßnahmen geplant oder verwendet.

# Quellen

**Mühlbacher, D. (2015):** Das Q in Agile, https://www.sogeti.de/globalassets/germany/download/veroffentlichungen-von-fachbeitragen/muehlbauer\_ots\_-agility\_15-4.pdf, Abruf: 15.10.2020