

ASE JavaScript

Training und Coaching für Scrum Teams



Passt: JavaScript und Agile Software Engineering

JavaScript (JS) ist mittlerweile sehr weit verbreitet als Sprache für (Web-)applikationen, Browser, kurz, für alle Anwendungen, die Nutzerinnen und Nutzern die verschiedensten Bedienelemente anzeigen. Gerade bei der Entwicklung mit JS bringt Agile Software Engineering große Vorteile. Beispielsweise deswegen, weil die Inhomogenität der verschiedenen Browser, Geräteklassen und Bildschirmgrößen automatisierte Tests sehr erstrebenswert macht. Typische Prinzipien des agilen Engineerings wie Test Driven Development (TDD) und Continuous Integration sichern die hohe Testabdeckung, die mühsames manuelles Herumprobieren erspart. Und natürlich gilt, dass Agile Software Engineering dabei hilft, technische Schulden zu vermeiden. Die entstehen schnell, weil man aufgrund der Dynamik der Plattform auch Hacks der Kategorie „quick & dirty“ in JS-Umgebungen einführen kann. Was kurzfristig vielleicht funktioniert, droht auf lange Sicht außer Kontrolle zu geraten - besonders in interpretierten dynamisch typisierten Sprachen wie JS, in denen kein Compiler überprüft, ob die verschiedenen Anwendungsteile grundsätzlich technisch zusammenpassen. Umso wichtiger ist handwerklich sauberes Vorgehen.“

ASE JavaScript

Dieser Kurs schliesst die Lücke zwischen dem Wunsch, nach Scrum zu arbeiten und den eventuell fehlenden handwerklichen Fähigkeiten im Programmieren mit JavaScript. Denn Scrum liefert ein hervorragendes Rahmenwerk dafür, wie sich die Entwicklerinnen und Entwickler organisieren. Wie sie es jedoch schaffen, innerhalb eines Sprints hochwertigen Code zu erzeugen, das ist nicht Gegenstand von Scrum. Oder wie sie den Code langfristig wart- und änderbar halten, damit die Entwicklungs-Kapazität frei bleibt für neue Features und nicht stärker und stärker vom Bugfixing geschluckt wird. Deshalb vermittelt dieses Training die dazu nötigen Prinzipien und Praktiken des Agile Software Engineering (ASE), hier speziell für JavaScript und AngularJS.

Das ASE Training wurde bereits 2010 von SAP und andrena entwickelt, zuerst für Java und ABAP. In der Zwischenzeit hat SAP mehr als 5500 Entwicklerinnen und Entwickler geschult. Das ASE Training wird seit 2011 auch für Teams außerhalb von SAP angeboten, zuerst für Java und ABAP, 2012 kam C# dazu und 2014 JavaScript.



Sie lernen als Team

- Praktiken, um die innere Softwarequalität langfristig auf hohem Niveau zu halten, z. B. Test Isolation, Test Planning, Refactoring
- den Umgang mit typischen Tools wie Mocha, Protractor, Jasmine, Bower ...
- Continuous Integration mit JavaScript zu verbinden und damit sicherzustellen, dass alle Bestandteile getestet und auslieferbar sind
- wie Sie JavaScript auch im Backend/Server einsetzen und so in allen Komponenten die gleichen Technologien verwenden können

Sie sind ein Team, das



- bereits mit JavaScript arbeitet und agil vorgehen will
- die vorhandenen Engineering-Skills in JavaScript erweitern will
- Agilität überzeugend findet und das Handwerkszeug lernen und vertiefen möchte, um mit JavaScript agil zu arbeiten
- sich bislang eher auf Designs konzentriert hat und lernen will, mit JavaScript professionell zu programmieren

ASE JavaScript

Training und Coaching für Scrum-Teams

Inhalte des Trainings

Der Kurs gliedert sich ein standardisiertes Training von einer Woche, anschließend begleiten unsere Trainer bzw. Coaches das Team in drei Sprints in die Projekte. Mit dieser Verbindung von Theorie und realer Projektpraxis fällt es leichter, die erlernten Techniken im Projektalltag umzusetzen.

Die Trainingswoche

In dieser Woche simulieren die Teilnehmenden ein Projekt, in dem sie eine Sudoku-Web-Applikation entwickeln. Die App unterstützt die Spieler beim Lösen der Rätsel. Beispielsweise lassen sich alle in einem Feld möglichen Ziffern anzeigen und aktualisieren, sobald weitere Felder ausgefüllt werden. Darüber hinaus können Spielende Hinweise anfordern, um nach strategischen Ansätzen bestimmte Ziffern zu platzieren oder zu streichen. Anhand dieses sehr anschaulichen und gleichzeitig anspruchsvollen Beispiels werden wichtige Programmier- und Scrum-Techniken eingeübt. Der Plan der Theorie-Woche umfasst folgende Lernmodule:

Monday	Tuesday	Wednesday	Thursday	Friday
Scrum (short refresh)	Test-driven development (Mocha)	Software Craftsmanship, Clean Code, Refactoring	Test Planning, UI Tests (Protractor)	Specification by Example / Acceptance Tests
Software Quality, "Done"				
Continuous Integration, Pair Programming	Sprint 1, 2 nd half	Sprint 2, 2 nd half	Sprint 3, 2 nd half	Sprint 4, 2 nd half
Introduction JavaScript / AngularJS				
Lunch break				
Introduction JavaScript / AngularJS	Unit Testing / Test Isolation	Asynchronous Testing	Exploratory Testing	Backlog Estimation, Release Planning
Sprint 1, 1 st half	Sprint 2, 1 st half	Sprint 3, 1 st half	Sprint 4, 1 st half	Q&A
End-of-day discussion				Retrospective

Die Sprint-Wochen

Sie verändern die Projektpraxis nachhaltig, weil sie die Möglichkeit bieten, die in der Theorie vermittelten Praktiken direkt auszutesten. Deshalb sind sie auch so wichtig für den dauerhaften Erfolg des Trainings. Alle unsere Trainer/innen sind selbst erfahrene Entwicklerinnen und Entwickler und kennen den Projektalltag aus der eigenen Berufspraxis.

Ein Training für Teams

Agiles Software Engineering ist eine Teamsache. Deshalb richtet sich dieses Training auch an das ganze Team und nicht nur an Einzelpersonen. Wir bieten das Training als Inhouse-Schulung an.