

INTERVIEW

„Lesen, coden und für Austausch sorgen“

Dieses Motto gibt Gregor Biswanger als Devise für angehende Entwickler aus.

Mangelndes Engagement kann man Gregor nun wirklich nicht vorwerfen. Er hat lange Zeit für die Microsoft-Technologien gebrannt. Trotzdem ist er dort nicht stehengeblieben. Jetzt lehrt er Node.js, Docker und JavaScript. dotnetpro hat mit dem Trackchair der Developer Week gesprochen.

Du bist auf der Developer Week der Trackchair für Frontend Architecture und zusammen mit Daniel Eichten für Container. Worauf hast du bei der Auswahl der Sessions besonders geachtet?

Gregor Biswanger: In einen Track passen nur sechs Sessions. Zur Auswahl stehen Hunderte von Einreichungen. Das führt zur großen Qual der Wahl, denn schließlich möchte man für jeden Teilnehmer etwas bieten. Die Themen sollten nicht nur einen Überblick geben, sondern auch für Einsteiger und fortgeschrittene Entwickler von Interesse sein. Meine Arbeit als Berater und Trainer bietet mir ein großes Spektrum an Themen. Ich versuche Sessions herauszupicken, die einen roten Faden bilden. Dann kommt der schwierige Teil der Auslese aus den bereits herausgesuchten Sessions: Welche könnte man eventuell weglassen und welche sind unentbehrlich?

Besonders wenn man von .NET kommt, stellt man sich ein Web-Frontend als eine wilde Sammlung von Spaghetti-Code-JavaScript-Bibliotheken vor. Stimmt das Vorurteil?

Biswanger: Tatsächlich ist das JavaScript-Chaos gar nicht mehr das Problem, wie es noch vor zehn Jahren mit jQuery und Co. der Fall war. Ganz im Gegenteil. Wir haben bereits seit zehn Jahren Angular, das sich bis heute im Enterprise-Umfeld stark etabliert hat. Es bietet eine Web-App-Entwicklung an, wie man sie von WPF gewohnt ist. Ich kenne einige Entwickler vom ehemaligen WPF-Core-Team, die von Microsoft zu Google gewechselt sind und jetzt beim Angular-Core-Team arbeiten. Zum Vorteil der .NET-Entwickler.

Welche Bedeutung hat die Architektur eines Frontends?

Biswanger: Das Frontend ist ein Teil der Softwarelösung und damit genauso wichtig wie die anderen Teile. Einige Anwendungen meiner Kunden sollen es ermöglichen, dass deren Kunden eigene Inhalte mit Logik hinterlegen können. Dann besagt die Microservice-Architektur, dass jeder Service eine eigene, unabhängige Web-App liefern soll. Am Ende möchte man dennoch alles unter einem Portal verbinden – beispielsweise unter Office 365 oder Amazon.

Es gibt so viele unterschiedliche Ziele mit harten Herausforderungen. Die richtig gewählte Architektur und die Herangehensweise sind deshalb essenziell für den Erfolg des Projekts. Die wichtigsten Aspekte der eben genannten Punkte werden auf der Developer Week 2020 gezeigt.

Du bist in vielen Technologien unterwegs. Wie kommt das?

Biswanger: Ich bin seit 19 Jahren im Microsoft-Universum aktiv. Ich war damals tierisch in Silverlight verliebt. Das war zum damaligen Zeitpunkt auch eine extrem gute Technologie, um die Grenzen des Browsers sprengen zu können. Als Silverlight dann inoffiziell beerdigt wurde, hat mich das getroffen. Ich habe in Redmond mit anderen Microsoft MVPs darüber gesprochen, die ebenfalls von Silverlight begeistert waren. Einige davon haben sich dann auf die native Webentwicklung mit JavaScript spezialisiert.

Dem bin ich gefolgt. Das war ein harter Brocken für mich. Mit der ASP.NET-Entwicklung konnte ich aber zumindest teilweise den C#-Komfort beibehalten.

Eines Tages habe ich von Intel ein Angebot aus San Francisco erhalten, ob ich Interesse hätte, die Architektur und Entwicklung eines Projekts zu betreuen. Allerdings hieß es, dass das ganze Backend auf Node.js und somit mit JavaScript läuft. Trotzdem wollte ich dieses Angebot unbedingt annehmen. Das Projekt ging mit einer wahnsinnigen Geschwindigkeit voran und ich war überrascht, wie erstaunlich gut die Qualität war. Weitere Beschäftigung mit JavaScript hat mich dann vollkommen überzeugt. So bin ich auf den Technologie-Blog des PayPal-Engineering-Teams [1] gestoßen. Das Team startete im Januar 2013 mit fünf Java-Entwicklern, um ihren neuen Auftritt zu programmieren. Im März 2013 wagte sich PayPal zusätzlich mit zwei Entwicklern an Node.js. Im Juni 2013 hatten beide Teams die Web-App mit der gleichen Funktionalität fertig. Das bedeutet: Sie haben mit weniger Entwicklern fast doppelt so schnell programmiert. Außerdem hatten sie beim Node.js-Projekt circa 33 Prozent weniger Code und rund 40 Prozent weniger Daten. Außerdem war die Performance deutlich höher als bei der Java-Lösung. Die gleiche Erfahrung habe ich bei meinen Projekten auch gemacht.

Einmal C#, immer C#. Ist das heute noch zeitgemäß?

Biswanger: Das lässt sich nicht allgemein festlegen, weil es auf die Wünsche jedes einzelnen Entwicklers ankommt. Klar, am liebsten möchten zahlreiche Entwickler in ihrem Ökosystem

bleiben. Das unterstützt auch Microsoft mit der Entwicklung von .NET Core, ML.NET, Xamarin, Blazor und so weiter. Mir gefallen die Konzepte von Microservices hier sehr gut. Die besagen, dass jeder Service mit der Technologie umgesetzt werden soll, die für die Aufgabe am besten geeignet ist.

Du bist Trainer, Sprecher auf Konferenzen, MVP und, und, und. Kommst du denn da überhaupt noch zum Coden?

Biswanger: Das Programmieren ist meine Leidenschaft, die ich regelmäßig zum Abschalten brauche. Ich arbeite nebenbei an ein paar Open-Source-Projekten. Ob abends im Hotelzimmer, auf der Zugfahrt zum Kunden oder im Homeoffice.

Stichwort eigene Projekte: Mit Electron.NET [2] hast du zusammen mit Robert Mühsig ein Projekt ins Leben gerufen. Was ist das und wofür kann ich es verwenden?

Biswanger: Auf das Projekt bin ich besonders stolz. In der JavaScript-Welt gibt es mit dem Open-Source-Projekt Electron eine tolle Lösung, mit der sich Cross-Plattform-Anwendungen für den Desktop entwickeln lassen. Microsoft selbst nutzt Electron für Visual Studio Code.

Robert ist der .NET-Welt treu geblieben und wir hatten überlegt, dass es möglich sein müsste, Electron unter .NET zu nutzen, sodass er keine Zeile JavaScript-Code schreiben muss. Nachdem wir alle technischen Möglichkeiten durchgegangen waren, bauten wir eine Art Brücke zwischen dem nativen Electron und der .NET-Welt. Als wir diese Beta-Version veröffentlichten, berichteten zahlreiche Entwickler-News über Electron.NET. Das Projekt hat mittlerweile über 4000 Stars auf GitHub und über 50000 Downloads bei NuGet. Daneben sind uns 280 Projekte gemeldet worden, die Electron.NET einsetzen. Einige davon sind sogar von Microsoft selbst. Eine Überraschung für uns war, dass ein Buch dazu veröffentlicht wurde. Das hatte ich zufällig über Amazon angeboten bekommen. Wir haben aktuell die Unterstützung für .NET Core 3 und Electron 7 implementiert.

Daneben hast du das Projekt Werner entwickelt.

Biswanger: Bei Projekt Werner geht es um meinen schwerbehinderten Bruder. Er ist gesund auf die Welt gekommen und erkrankte in seinem dritten Lebensmonat an Gehirnentzündung. Seitdem hat er eine spastische Behinderung und sitzt im Rollstuhl. Er ist komplett eingeschränkt, kann nicht selbst essen, trinken, auf die Toilette gehen oder deutlich sprechen.

Ich hatte damals von Intel kostenfrei eine 3D-Kamera zum Herumexperimentieren erhalten. Das Interessante daran war, dass diese bestimmte Gesichtsgesten erkennen konnte. Das brachte mich auf die Idee, einen Prototyp in WPF zu entwickeln, der es schwerbehinderten Menschen ermöglichen sollte, mit anderen Menschen zu kommunizieren. Es wird eine virtuelle Tastatur eingeblendet und mit Mundgesten ein Cursor bewegt. Anschließend kann man den geschriebenen Text vorlesen lassen. Inspiriert wurde ich dazu von Stephen Hawkings Lösung. Als mein Bruder den Prototyp ausprobiert hat, hat es tatsächlich geklappt. Er konnte das erste Mal in seinem Leben etwas selbstständig kontrollieren. Sein glückliches Gesicht zu sehen war ein sehr bedeutender Moment in

● Gregor Biswanger

Gregor ist Microsoft MVP, Intel Black Belt, Intel Software Innovator und freier Berater, Trainer, Autor sowie Sprecher. Er berät große und mittelständische Unternehmen, Organisationen und Agenturen rund um die Themen Softwarearchitektur, Cloud-, Web- und Cross-Plattform-Entwicklung mit C# und JavaScript/TypeScript. Seine Schwerpunkte sind der MEAN-Stack, ASP.NET-Core, Docker, Azure, Domain-Driven Design und Microservices. Außerdem schreibt er für Heise.de und Artikel für Fachmagazine wie die dotnetpro. Er ist international als Sprecher auf Konferenzen und in Communities unterwegs. Sie erreichen ihn unter <http://about.me/gregor.biswanger>.



meinem Leben. Dieses Projekt habe ich anschließend als Open Source veröffentlicht und einen Hackathon organisiert, bei dem mich Entwickler ein Wochenende dabei unterstützen, die Software voranzubringen. Aber ich brauche Unterstützer, alleine schaffe ich das nicht.

Wie viele Anfragen von Headhuntern erhältst du pro Woche?

Biswanger: Viel zu viele, und keiner von ihnen schaut mein Profil vorher ernsthaft an.

Welchen Rat würdest du einem Nachwuchsentwickler geben?

Biswanger: Lesen, coden und für Austausch sorgen. Es gibt meisterhafte Standardwerke namhafter Autoren, die technologieunabhängig die wichtigsten Aspekte vermitteln, etwa „Clean Code“, „Clean Coder“ und „Clean Architecture“. Dann gibt es zahlreiche Coding-Übungen, die Code Katas heißen. Die dotnetpro hat zum Beispiel immer ganz tolle Übungen drin. Seht euch an, was sich hinter Test Driven Development verbirgt, und löst damit die Code-Übungen. Dann gibt es deutschlandweit ein Dutzend User Groups und Meetups, wo ihr euch mit Gleichgesinnten austauschen könnt.

Zum Schluss: Was fällt dir spontan zu diesen Begriffen ein?

JavaScript: Die am häufigsten missverstandene Programmiersprache der Welt.

TypeScript: Produktivitäts-Booster.

.NET Core: Hat auch einige Konzepte von Node.js übernommen, und das gefällt mir!

Cross-Plattform: Ganz klar! Electron.NET!

Projekt Werner: Glücklicherweise, meinen Bruder am Leben teilhaben zu sehen.

Vielfalt: Ein wichtiges Thema für stabile Systeme. ■

[1] PayPal Engineering,

<https://medium.com/paypal-engineering>

[2] Elektron.NET, www.dotnetpro.de/SL2004DWX20_1