MATERNA
Information & Communications

Weiterbildung und Wissenstransfer durch Fachvorträge von Materna-Experten

Die IT ist eine Branche, die sich immer weiterentwickelt und die von neuen innovativen Technologien und Möglichkeiten lebt. Unsere erfahrenen Experten sind immer auf dem neuesten Stand und teilen dieses Wissen gerne bei Fachvorträgen mit den Studenten. Die Vorträge können in den Hochschulen als Präsenz-Veranstaltung, digital per Live-Stream, Webcast oder auch an einem unserer Standorte stattfinden. Hier finden Sie Themenbeispiele für Fachvorträge. Wenn Sie mehr zu einem Vortrag wissen möchten oder Interesse an weiteren Themen haben, schreiben Sie einfach an: hochschulen@materna.de.

Dr. Johannes Neubauer

 # Do Less
mit Agile GitOps blitzschnell in die Produktion

Was tun, wenn Mitarbeiter ungerne mit den Software-Applikationen arbeiten, die ihnen im Arbeitsalltag zur Verfügung stehen? Oder wenn Endkunden schlechte Bewertungen in den App Stores abgeben? In dieser Situation sind immer mehr Unternehmen. Doch wie sollen sie reagieren, ohne dass die Kosten in der Anwendungsentwicklung explodieren? Mit Agile GitOps werden sie blitzschnell produktiv in der Applikationsentwicklung.

Titel: „Cloud == 7?“

Die Cloud ist mittlerweile allgegenwärtig. Doch ist Cloud Native gleichbedeutend mit „Wolke 7“? Was bedeutet es „in der Cloud“ zu sein und wie kommt man dort hin? Was ändert sich in der Cloud für meine Art zu entwickeln? Wo beginnt sie und wo hört sie auf? Sollte ich vielleicht sogar in der Cloud entwickeln? Was sind die Voraussetzungen für die Cloud? Kann ich als Student bereits in die Cloud?

 # Heiter, weil wolkig!
Cloud-Software Development mit AWS: Multi-Mandantensysteme werden serverless

In großen Organisationen oder auch als Anbieter von SaaS-Lösungen kommt unweigerlich der Moment, an dem ein System zu einem Multimandantensystem wird. Dabei entsteht eine neue Dimension der Komplexität: die Produktlinie. Es müssen Varianten angeboten werden, um die Anforderungen aller Beteiligten zufrieden zu stellen. Das macht einerseits die Umsetzung, Qualitätssicherung, Monitoring, Pflege und das Deployment komplizierter. Andererseits steigen im Normalfall die Betriebskosten und die vertikale Skalierung ist endgültig am Ende mit „ihrem“ Latein. In diesem Vortrag zeigen wir, wie Agile GitOps, Cloud Native und Serverless Computing in Kombination helfen, diese Probleme in den Griff zu bekommen.

Über Dr. Johannes Neubauer:
Johannes ist Senior Projektmanager und Teamleiter mit Erfahrung sowohl in Research & Development als auch einer großen Bandbreite an Industrie- und Kundenprojekten. In seiner Karriere hat er erfolgreich die verschiedensten Rollen wie Consultant, Anforderungsmanager, Entwickler, Tester (im Bereich Testautomatisierung), Architekt, Forscher und Projektmanager in Projekten von Desktop-Applikationen über Enterprise (Web) Applikationen, Microservices und Cloud Native Anwendungen bis zu mobilen Apps eingenommen.

Frank Pientka:

DevSecOps aber sicher! Kontinuierlich sicherer entwickeln

Agile Softwareentwicklung hat sich in den letzten Jahren weiterentwickelt. Mit DevSecOps steht der nächste größere Schritt an, um schneller auf Änderungen oder Sicherheitslücken reagieren zu können. Der durch DevSecOps eingeleitete Kulturwandel (shift left) ist ein wichtiger Baustein auf dem Weg in die Cloud. Nur wer DevSecOps wirklich lebt, kann Cloud-native Services kontinuierlich liefern und verbessern. Es wird eine Einordung von DevSecOps und der dabei verwendeten Methoden gegeben. Es wird eine typische Werkzeugkette vorgestellt und Vor- und Nachteile der verschiedenen Ansätze werden diskutiert.

Alles nur ge-cloud? Architecting for the Cloud Best Practices

Cloud-Computing ist inzwischen eine normale Option geworden. Unabhängig vom jeweiligen Cloud-Anbieter gibt es einige Prinzipien und Muster zu beachten. In dem Vortrag wird das Vorgehen eines Cloud Readiness Checks vorgestellt. Dieser betrachtet die folgenden Gebiete und ordnet diese in die bestehende Cloud-Strategie ein:

1. Prozesse: Feststellung des Cloud-Reifegrades, Migrations-/Modernisierungsstrategien, Poly-Cloud, Cloud-Center-of-Excellence, Cloud-first-Strategie

2. Kosten: Optimierung der Kosten (Right-sizing), passende Dienste für jetzt und später ("so viel wie nötig", "darf es etwas weniger sein")

3. Betrieb: Verbesserung des Monitorings, Deployment (Leistung, Effizienz), Automatisierung, Transparenz, Feedback, Governance, Einhaltung der SLAs

4. Entwicklung: Verbesserung der Geschwindigkeit, Aufwände, Qualität, DevOps, Infrastructure-as-Code, PaaS, CaaS, FaaS

5. Architektur: Wahl der passenden Dienste, Architekturmuster zu den angestrebten Qualitätszielen

6. Sicherheit: Risikoerfassung, -Minderung, Erhöhung des Schutzes, Ausfälle minimieren, Auswirkungen von Angriffen minimieren, Reaktionsfähigkeit erhöhen

So hilft der Cloud Readiness Check, den eigenen Weg stressfrei in die Cloud zu finden und Risiken frühzeitig zu adressieren.

Von der Software als Kostenfaktor zum Innovations- und Wachstumstreiber - Was versteht man unter Plattformökonomie und wie geht das technisch?

Die Wartungsaufwände und Softwarekosten steigen nicht nur wegen neuer Anforderungen. Deswegen geht der Trend immer mehr hin zu Softwareökosystemen. Dabei bilden sich oft Zusatzdienste und -Produkte auf der Basis einer Plattform. Da man es bei der Entwicklung solch einer Plattform mit einer Vielzahl von Stakeholdern zu tun hat, stellt das neue Herausforderungen an den Bau, den Betrieb und die Dokumentation solch eines Systems. Damit mit APIs Geld verdient werden kann, sind einige Voraussetzungen nachhaltig zu erfüllen. Es werden erfolgreiche Plattformökosysteme und deren Voraussetzungen vorgestellt. Letztendlich werden immer mehr traditionelle Unternehmen dadurch zu Softwareunternehmen.

DevSecOps: Open Source Tools & Continuous Development

Link zum Vortrag:
https://www.youtube.com/watch?v=KeVds-FuJnU

Über Frank Pientka:
Frank ist Principal Software Architekt bei Materna in Dortmund und Gründungsmitglied der iSAQB, die sich um die Zertifizierung von Software-Architekten kümmert. Er beschäftigt sich seit mehreren Jahrzehnten mit skalierbaren und flexiblen Diensten in der Cloud. Dabei setzt er sich intensiv mit innovativen Software-Themen auseinander. Darüber schreibt und referiert er über verschiedene Kanäle.

Stefan Pauka

Less Code via Convention: Dank Code-Generierung mehr Zeit für die wichtigen Dinge im Leben

Wenn Softwareanwendungen geschrieben werden, geht es neben vielen fachlichen Fragen auch um das Ausarbeiten und Implementieren technischer Schnittstellen. Die Anbindung einer Webapp ist oft eine ungeliebte, eintönige Fleißaufgabe. Wie schön wäre es, wenn bereits Implementierungen der Schnittstellen in Service-Anwendungen reichen würden, um den „langweiligen" Teil des Boilerplate-Codes für solche Clients zu generieren?

In diesem Vortag wird gezeigt, wie Web-Services direkt zur Generierung von API-Clients verwendet werden können, und wir stellen uns die Frage, ob wir uns nicht häufiger in kleinem Rahmen einen eigenen Generator gönnen sollten.

Abschließend werfen wir noch einen Blick in eine Single Page Application: Häufig finden sich hier Frameworks wie Redux zum Verwalten eines globalen States, die gern zu großen Anteilen von Boiler-Plate Code einladen. Für diese kann man sich mit TypeScript viel Standard-Tipparbeit abnehmen lassen, ohne in komplizierte Typdefinitionen zu verfallen.

Ultra Hybrid Apps

Was tun, wenn Mitarbeiter ungerne mit den Software-Applikationen arbeiten, die ihnen im Arbeitsalltag zur Verfügung stehen? Oder wenn Endkunden schlechte Bewertungen in den App Stores abgeben? Egal ob B2B, B2C oder B2E, die Nutzer von Applikationen sind mittlerweile verwöhnt. Neben einer Webapp soll es schon auch eine native App geben und schön gestaltet sowie intuitiv zu bedienen sowohl für iOS- als auch Android-Nutzer soll sie sein. Viele Unternehmen sind daher in dieser Situation. Doch wie sollen sie reagieren, ohne, dass die Kosten in der Anwendungsentwicklung explodieren? Der Vortrag zeigt auf, welche Möglichkeiten Ultra Hybrid Apps für diese Herausforderung bieten.

Über Stefan Pauka: Stefan arbeitet als Softwareentwickler bei Materna in agilen Projekten. In diesen entwickelt er Lösungsstrategien für die Digitalisierung geschäftskritischer Prozesse und setzt diese im Team um. Innerhalb dieser Projekte programmiert er Webservices in Java, C# und Javascript. Im Web realisiert er Single-Page-Anwendungen mit Vue.js, React und Dojo 2.0 – am liebsten mit TypeScript. Neben seinem Faible für neue Herausforderungen ist Stefan nicht zuletzt dank TypeScript und Visual Studio Code bekennender Microsoft-Fan. In seiner Freizeit programmiert er am Hobbytisch in kleinen Projekten wie einem Phaser.IO-„Einparktraining".

Georg Wolffgang

Modellbasierte Entscheidungsfindung

Welcher Mitarbeiter wird als nächstes mit der aktuellen Aufgabe betraut? Ist die Annahme eines Projektantrages wirtschaftlich sinnvoll oder sprengt er womöglich unser Budget?

In Geschäftsprozessen gibt es am laufenden Band Entscheidungen zu treffen. Falls es ein Regelwerk gibt, nach dem sich die Fragestellungen eindeutig beantworten lassen, kann diese Logik auch von einem Computer automatisiert übernommen werden. Aber wie bringt man einem Computer bei, Entscheidungen geschickt zu treffen? Welchen Einfluss hat dies auf den Geschäftsprozess?

In dem Vortrag wird die Modellierungssprache Decision Model and Notation (DMN) an Beispielen vorgestellt. Außerdem soll das Interesse für angewandte Modellierung geweckt werden. Der Vortrag wird durch interaktive Beispiele gestützt. Diese ermöglichen es, die Thematik bei geringerer Teilnehmerzahl auf spielerische Art und Weise in Workshops selber zu entdecken.

Über Georg Wolffgang: Georg ist Software Developer und Business Consultant sowie Grund-, Haupt- und Realschullehrer für Mathematik und Naturwissenschaften. Nach mehrjähriger kinderorientierter Lehrtätigkeit orientierte er sich zur Erwachsenenbildung hin neu. In der Abteilung Software Factory der Business Line Public Sector nahm er nicht nur erfolgreich die Rollen des Entwicklers und Testers sowie Dozenten für firmeninterne Traineeprogramme ein, sondern gewann ebenfalls erste Erfahrungen als Business Consultant und Anforderungsmanager.

Daniel Schmitz-Hübsch

Decision Requirements Level: Das Schlüsselelement von DMN

Die Decision Model and Notation (DMN) unterscheidet zwischen dem Decision Requirements Level (DRL) und dem Decision Logic Level (DLL). Insbesondere zu Beginn der Entscheidungsmodellierung sind das DRL und das dazugehörige Decision Requirements Diagram (DRD) von zentraler Bedeutung, da sie die Grundlage für die Modellierung bilden. Dieser Vortrag zeigt, welche Elemente eines DRD für Business-Analysten wichtig sind und wo Schwierigkeiten auftreten können. Darüber hinaus wird gezeigt, wie Business-Analysten in konkreten Projekten bei der Erstellung der DRL vorgehen.

Entscheidungsmodellierung mit integrierten Tests

Bei der Erstellung komplexer Entscheidungsmodelle ist das Qualitätsmanagement von besonderer Bedeutung. Dazu zählt auch, dass die Business-Analysten frühzeitig in der Lage sind, ihre Entscheidungsmodelle auszuführen und zu testen.

Dieser Vortrag zeigt, wie Testfälle bereits im Rahmen der Entscheidungsmodellierung von Business-Analysten erstellt und ausgeführt werden können.

Über Daniel Schmitz-Hübsch: Daniel ist Software-Entwickler mit dem Fokus auf Anwendungen in der öffentlichen Verwaltung. Er hat einen Master-Abschluss in Wirtschaftsinformatik mit Schwerpunkt auf der mobil-orientierten Analyse von Geschäftsprozessen. Seit über fünf Jahren beschäftigt er sich mit der Modellierung und technischen Umsetzung von Geschäftsprozess- und Entscheidungsmanagementsystemen. Als Softwareentwickler ist er verantwortlich für die Entwicklung von hochverfügbaren Entscheidungsapplikationen unter Verwendung von Rule Engines wie IBM Operational Decision Management oder RedHat Drools.

MATERNA Information & Communications

Materna Information & Communications SE

Voßkuhle 37
D-44141 Dortmund

Tel: +49 231 5599-00
sales@materna.de
hochschulen@materna.de

© Materna 2021