

Medienmitteilung vom 13. September 2023

Swiss Low Code Day 2023: Die Digitalisierung beschleunigen

Low-Code-Ansätze erlauben, Geschäftsanwendungen und Automatisierungen durch visuelles Zusammenfügen vorgefertigter Software-Bausteine zu erstellen und mittels wenig Programmcode – dem «Low Code» eben – individuell anzupassen. Low Code wirkt also als Enabler einer beschleunigten bzw. vereinfachten Softwareentwicklung, die dann je nach Produkt auch von Nichtinformatikerinnen und Nichtinformatikern vorgenommen werden kann. Diese sogenannte Citizen Developer sind in der Regel auf Fachseite angesiedelt – Digitalisierungsvorhaben stützen sich damit in der Organisation viel breiter ab als bis anhin. Doch wie schlägt sich Low Code tatsächlich in der Praxis, und wie gut funktioniert Citizen Development in der Realität?

Der am 16. November in Zürich stattfindende **Swiss Low Code Day 2023** gibt Antworten auf diese Fragen. Er ist vielfältige Plattform sowohl für die Anwender- als auch Anbieterseite; auf Teilnehmende warten Ein- & Ausblicke, Erfahrungsberichte und Showcases rund um die Themen Low Code und Citizen Development. Das Programm für den Swiss Low Code Day haben das Institut für Wirtschaftsinformatik der Universität St.Gallen (IWI-HSG) und das an der OST – Ostschweizer Fachhochschule angesiedelte LowCodeLab@OST gemeinsam gestaltet. Als Premiumsponsoren sind Zühlke und SAP mit an Bord; unterstützt wird die Veranstaltung zudem von Isolutions, Plain IT, Flowable, Simplifier, BOC Group, HCL Software und Valantic.

Wann: 16. November 2023, 13:30 Uhr - 20:00 Uhr

Wo: FFHS Gleisarena, [Zollstrasse 11, 8005 Zürich](#) (direkt beim Zürcher HB)

Infos, Programm & Anmelden: <https://www.lowcodelab.ch/swisslowcodeday2023>

Für Rückfragen:

- Prof. Dr. Rainer Endl, Professor für Wirtschaftsinformatik, Co-Leiter LowCodeLab@OST, rainer.endl@ost.ch
- Dr. Christoph Baumgarten, Dozent für Wirtschaftsinformatik, Co-Leiter LowCodeLab@OST, christoph.baumgarten@ost.ch
- Michael Breu, Kommunikation OST, 058 257 44 66, michael.breu@ost.ch