

A-VIS - Visualisierungssoftware



Was ist A-VIS?

Einfach, intuitiv und grafisch ansprechend ist unser Anspruch.

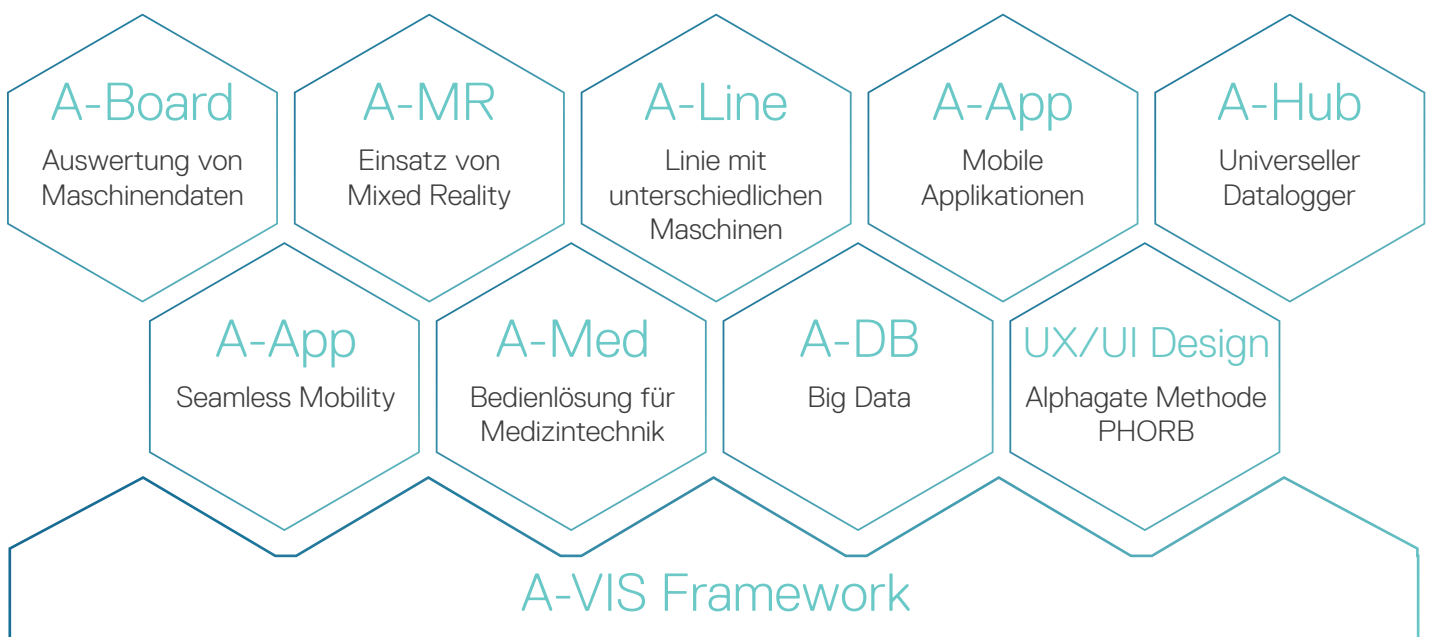
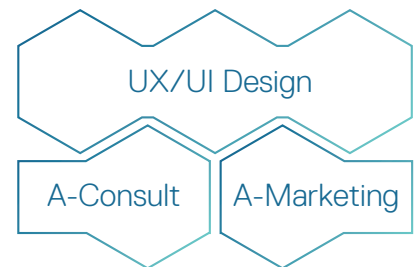
Innovativ und technologisch nachhaltig in der Umsetzung ist unser Leitsatz.

Inspiziert von der Vertrautheit der Natur entwickelt Alphagate ein einzigartiges UX/UI Design. Mit A-VIS wird dieses in eine user- und prozessorientierte Software gegossen.

Alphagate entwirft und realisiert komplette Bedienkonzepte für Maschinenbauer und Hersteller medizinischer Geräte. Das individuell erstellte UX/UI Design spiegelt die Anforderungen des Prozesses und die bestmögliche Anleitung und Unterstützung für den Betreiber des Geräts wider. Alphagate verwendet seine eigene SCADA-Software A-VIS, um ein einzigartiges

Software-Framework zu erstellen, das auf dem Bedienkonzept basiert. A-VIS bietet Konnektivität zu allen wichtigen SPS und Datenquellen. Alphagate bietet eine Reihe zusätzlicher Erweiterungen und Produkte an, die nahtlos mit der Alphagate-Lösung verwendet werden können, wie A-MR (Mixed Reality), A-DB (Big Data) und viele mehr.

Leistungen



Aufbau A-VIS

A-VIS ist ein Visualisierungstool, erstellt und laufend optimiert für die Erstellung eines HMI für den Maschinenbau, Anlagenbau und Medizintechnik. Es ist für die Kommunikation zwischen Mensch und Maschine ausgelegt.

A-VIS ist ein innovatives Werkzeug zum Erstellen von intelligenten Visualisierungs- und Bedienlösungen, die den allerhöchsten Ansprüchen genügen. Entwickelt von Software- und UX Experten für den internationalen Einsatz an Produktionsmaschinen, Anlagen und Geräten der Medizintechnik. A-VIS ist mit purem Java umgesetzt und mit einer durchgängigen Client-Server-Architektur ausgestattet. Der Funktionsumfang entspricht dem eines modernen HMI oder SCADA Werkzeuges, das zusätzlich durch Anforderungen aus der Usability und Internationalisierung ergänzt wurde.

A-VIS bildet derzeit die Plattformen Java AWT, Java Swing, Java Fx und HTML5 ab. Somit sind die Applikationen, die mit dieser Entwicklungsumgebung projektiert wurden, auf allen gängigen Plattformen lauffähig – von kleinen Singleboard Lösungen über klassische IPC´s unter Windows oder Linux bis hin zu Smart Devices. Es wurde bei der Entwicklung besonderes Augenmerk auf kundenspezifische Anpassungen gelegt. Selbstverständlich wird die Entwicklungsumgebung mit dem speziell für den Kunden ausgearbeiteten Styleguide und Bedienkonzept ausgeliefert – der Styleguide ist quasi in der IDE integriert.

Offen für alles

A-VIS bietet Schnittstellen für alle gängigen Steuerungen und Datenquellen.

Module und umfangreiche Funktionalitäten

Message Manager

Der Hauptzweck des Message-Managers ist das Abrufen, Verarbeiten und Weiterleiten von Nachrichten an andere Module. Der Manager bietet grundlegende Funktionen, um die Hauptanforderungen einer Anwendung in Bezug auf Nachrichten zu unterstützen.

Page/View Manager

Der Hauptzweck des A-VIS Page-Managers besteht darin, die Verwaltung einer Hierarchie von Seiten und Vorlagen zu vereinfachen.

Datalogger Manager

Der Manager sammelt Daten aus verschiedenen Quellen, Steuerelementen und Datenbanken, die zur Analyse und Diskussion zur Verfügung stehen.

Unit Manager

In vielen Fällen muss ein HMI Werte in verschiedenen Einheiten anzeigen. Der Unit-Manager deckt diese Bedürfnisse ab.

Dataset/Recipe Manager

Der Manager ist ein Mechanismus zum Speichern und Wiederherstellen von Geräteeinstellungen und Rezepten.

Text Manager

Für alle textbezogenen Anforderungen stellt A-VIS den Text-Manager zur Verfügung, der die Funktionalität enthält, um alle Anforderungen zu erfüllen.

User Manager

Der Umgang mit Maschinen ist komplex, häufig sollten Aufgaben unterschiedlichen Benutzerrollen zugeordnet werden.

Graphics Manager

Der Zweck des Managers besteht darin, die Verwaltung von Grafikressourcen zu vereinfachen, indem Funktionen bereitgestellt werden, die die erforderlichen Dateioperationen im Hintergrund ausführen.

Color Manager

Farben sind in HMIs wichtig. Der Manager kümmert sich um unterschiedliche Wahrnehmungen, Kulturen und Situationen.