INFINITE VELOCITY MATURITY FRAMEWORK

CREATING SOFTWARE THAT SHAPE TOMORROW

	FOUNDATION Developing on a CD 3.0 platform, but the cycle is poorly automated.	NOVICE Working with basic automation on a reactive level.	INTERMEDIATE Running average CD 3.0 technologies with proactive elements.	ADVANCED Working with advanced CD 3.0 tech, that is quantitively managed.	EXPERT Increasingly utilizing AI to improve the CD 3.0 development cycle.
INTELLIGENCE Making business decisions based on gathered user data.	Tracking customer behaviour on a server and receive feedback.	Basic monitoring of app usage and handling customer feedback.	Advanced customer monitoring and A/B testing in place.	Receiving predefined metrics and reports. Decisions being made based on detailed analytics.	Real-time data collection analysis and reporting using AI.
PLANNING Automating backlog item creation and prioritization to improve collaboration.	Managing the complete backlog on a centralized server.	Managing all work by means of digital backlog.	Automatically creating items for the backlog.	Automatically receiving backlog prioritization suggestions.	Automatically receiving backlog prioritization suggestions.
CONTINUOUS ENGINEERING The evolutionary development and short-cycle delivery of software	Engineering software in a monolithic system.	Engineering truly agile software in an encapsulated, layered architecture.	Engineering software in a microservice architecture.	Engineering software with cloud-based design patterns and services.	Engineering serverless software in a cloud-based architecture.
INTEGRATION Automatically building your software to shorten the development cycle.	Running a centralized version control system. Running a centralized build server.	Running a workflow ochestrator and receiving reports. Running builds while the company is asleep.unning a workflow ochestrator and receiving reports.	Triggering builds after the commit of a new feature.	Running integrations on a scalable microservice architecture. Staged Integrations - compiling the source code that was edited only.	Automatically scaling continuous integration services.
TESTING Automatically testing newly developed features to avoid tedious work.	Running a centralized unit-test server. Manually starting unit tests.	Running unit tests in a Continuous Delivery pipelines. Manually starting your automated integration tests.	Triggering integration tests in your Continuous Delivery pipeline. Manually starting your automated acceptance-test.	Triggering acceptance tests in your Continuous Delivery pipeline. Manually starting your automated security and performance tests.	Triggering end-2-end regression tests in your pipeline.
DEPLOYMENT Automatically deploying new builds to scalable environments.	Running a centralized deployment server. Running basic deployment.	Running basic deployment scripts. Automatically deploying to a test server after a succesful build.	Automatically deploying to the production server using a pipeline.	Automatically deploying without any downtime.	Automatically deploying on endless scalable platforms.

wiconic.com | info@wiconic.com | +31(0)30 - 268 53 98