

1 Was ist Circle CI?

Circle CI ist eine Plattform für kontinuierliche Integration, die sowohl als gehostet als Software as a Service oder selbst gehostet verwendet werden kann. Circle CI unterstützt dabei eine große Auswahl an gängigen Sprachen und Umgebungen mit Beispielprojekten mit ausführlicher Dokumentation, an denen sich der Einsatz des Tools auch für CI-Anfänger einfach erlernen lässt.

1.1 Was ist Kontinuierliche Integration?

Die Idee hinter Kontinuierliche Integration ist, die einzelnen Komponenten einer Software während derer Entwicklung immer wieder zu einer Einheit zusammenzufügen. Üblicherweise beinhaltet dieser Vorgang nicht nur das Bauen des gesamten Codes, sondern auch das Ausführen von Tests, die sicherstellen sollen, dass sich durch die neu hinzugekommenen Änderungen keine Fehler in den Code eingeschlichen haben. Das Ziel dieses kontinuierlichen Bauens und Testens ist, die Qualität der entwickelten Software zu erhöhen, indem man auftretende Fehler frühzeitig bemerkt und verbessert.

Um ein eigenes Projekt kontinuierlich integrieren zu können, müssen ein paar Grundvoraussetzungen erfüllt sein. Der erste wichtige Schritt ist eine gemeinsame Codebasis, an der ein oder mehrere Entwickler arbeiten können. Eine Möglichkeit dafür sind Versionsverwaltung wie GIT oder SVN, in die alle Entwickler ihre Änderungen am Code einpflegen können. Eine weitere Voraussetzung ist die Möglichkeit, den Code automatisch bauen zu lassen. Für die meisten Sprachen gibt es hierfür Build-Tools, in denen die einzelnen Schritte des Übersetzens beschrieben werden können, um diese später automatisiert auszuführen. Beispiele wären hierfür Make oder CMake für C und C++ Projekte oder Maven für Java Projekte. Diese übernehmen nicht nur das Übersetzen des eigenen Codes, sondern auch das Anbinden an bereits bestehende Projekte, die als Abhängigkeiten definiert werden. Ziel hierbei ist es am Ende, die gesamte Software im besten Fall durch einen Befehl bauen zu können, ohne dass weitere Schritte von Hand notwendig sind. Eine weitere wichtige Voraussetzung ist es, Tests zu schreiben, mit denen sichergestellt werden kann, dass die Änderungen richtig funktionieren und nicht in bestehendem Code zu Fehlern führt. Hier können Tools wie SonarQube verwendet werden, die automatisch prüfen, welcher Anteil des Codes durch Tests bereits abgedeckt sind. Nach erfolgreicher Integration kann beispielsweise das Überspielen der Software in eine Produktivumgebung erfolgen. Hierbei wird dann von Continuous Delivery, also kontinuierlichen Auslieferung der Software gesprochen.

Auch wenn Kontinuierliche Integration gerade am Anfang mit zusätzlichem Aufwand verbunden ist, gibt es auf lange Sicht doch viele Vorteile, die den Einsatz in den meisten Softwareprojekten sinnvoll machen. Tauchen bei der Entwicklung Fehler auf, so können diese schnell und zeitnah behoben werden. Zudem ermöglicht es kontinuierliche Integration, immer eine sehr aktuelle Version der Software für Demonstrationen für beispielsweise Vertriebszwecke zur Verfügung zu haben. Zudem wird durch kürzere Integrationszyklen auch der Aufwand der einzelnen Integrationen der Neuerungen in den Produktionscode deutlich verringert. Die Gefahr das neu entwickelte Komponenten nur schwer in die bestehende Code Basis integriert werden können, wird dadurch minimiert.

1.2 Welche Features bietet Circle CI für Entwickler?

CircleCI bietet eine Reihe von interessanten Features für Entwickler. Hier eine kurze Zusammenfassung:

SSH Verbindung zu Build Slave Über eine Interactive Login Shell können fehlgeschlagene Tasks leicht untersucht werden

Paralleles Testen auf mehreren Maschinen Eine größere Menge an Tests können auf mehrere Maschinen verteilt werden

Konfigurierbare Zuweisung von Ressourcen Die Belastung für CPU und RAM pro Job können konfiguriert werden

Caching von Daten Die Laufzeit von Tests kann durch das Cachen von Daten verbessert werden

Circle CI Workflows Über Workflows können verschiedene Aufgaben koordiniert werden

1.3 Welche Features bietet Circle CI für Betreiber der Software?

Auch für die Betreiber der eigenen Software hält Circle CI eine Reihe interessanter Features bereit.

Monitoring Einfache Überwachung der installierten Umgebung

Nomad Cluster Nomad als Framework für die Orchestrierung der Container Umgebungen

Rest APIs Alle Funktionen können über eine Rest Schnittstelle angestoßen werden

Einfache Fehlersuche für eigene Installationen Einfache Fehlersuche bei Installationen von Circle CI auf eigenen Servern

Einfache Übersicht Dashboard mit der Übersicht die einzelnen Repositories und deren Ergebnisse

2 Einführung in die Verwendung von Circle CI

Der wichtigste Punkt an einem neuen Tool für uns als Entwickler ist es, dass wir es effektiv und effizient einsetzen können. In unserem Fall stellt sich damit die Frage, wie wir ein neues oder bereits bestehendes Projekt kontinuierlich integrieren können. Hierbei sehen wir uns zuerst die bereits gehostete Version von Circle CI an, da diese uns gerade am Anfang administrativen Aufwand erspart. Für die Integration des eigenen Projekts in Circle CI macht es im Grunde genommen keinen Unterschied, ob Code für das Projekt bereits geschrieben wurde, oder ob man mit einem neuen Projekt beginnt, in dem es weder Code noch Tests gibt. Das Projekt benötigt lediglich eine Datei config.yml in einem Ordner .circleci, den man in das Root Verzeichnis des Projektes ablegt, das in Circle CI gebaut werden soll.

3 Vergleich zu anderen Tools und Anbieter

Platzhalter