

Weboberfläche für Portfoliomanagement

Studiengang: BSc in Informatik | Vertiefung: IT Security
Betreuer: Prof. Dr. Kai Brännler
Experte: Alain Joray

Die bereits bestehende Software für Portfoliomanagement des Praxispartners Spiniceps AG, welche für den automatisierten Handel von Kryptowährungen wie Bitcoin ausgelegt ist, soll durch eine benutzerfreundliche webbasierte Oberfläche ergänzt werden, sodass Kunden sich registrieren, ihre Zugangsdaten für die Börse hinterlegen und die Funktionalität der darunterliegenden Software nutzen können.

Übersicht

Bitcoin ist ein innovatives Zahlungsnetzwerk sowie eine neue Art von Geld. Im Gegensatz zu Fiatwährungen wie der Schweizer Franken, wird Bitcoin nicht von einer zentralen Stelle ausgegeben und verwaltet, sondern in einem ausgeklügelten Peer-to-Peer Netzwerk von allen Teilnehmern gleichermaßen kontrolliert. Da Bitcoin eine verhältnismässig junge Assetklasse ist und verglichen mit anderen Assets über wenig Handelsvolumen verfügt, braucht es relativ wenig Geldmittel, um den Preis von Bitcoin oder anderen Kryptowährungen um mehrere Prozentpunkte auf oder ab zu bewegen. Diese Preisvolatilität macht sich die Software der Spiniceps AG zunutze, indem automatisiert Kauf- und Verkaufsaufträge angelegt werden, um damit Gewinne zu erzielen.

Ziel der Applikation

Die bestehende Software, welche bisher nur technisch versierten Personen über ein Kommandozeilentool zugänglich war, soll durch eine benutzerfreundliche Weboberfläche erweitert werden, um Bitcoin und andere Kryptowährungen auf der derzeit unterstützten Handelsbörse Bitfinex automatisiert zu handeln. Der Kunde kann demnach für alle seine Trading Pairs (z.B. BTC:USD) einen sogenannten Orderleiter-Maintainer nach seinen Wünschen konfigurieren, um Kauf- und Verkaufsaufträge anzulegen. Ausserdem ist er in der Lage, das Handeln auszusetzen und die Orders zu löschen, zu bearbeiten und später wieder zu erstellen.

Umsetzung

Als Frontend-Technologie wurde das Javascript Framework Vue.js gewählt, während das Backend auf dem Python Webframework Flask basiert. Für das Backend wurde Python gewählt, da Funktionen der bestehenden Software, die ebenso in Python geschrieben sind, direkt verwendet werden konnten. Als Authentifizierungsmechanismus werden JSON Web Token (JWT) verwendet, da sie den Grundsatz der Zustandslosigkeit von REST erfüllen und das Backend dadurch

auch skalierbarer wird. Die ganze Kommunikation des Frontends geschieht über das Backend, welches wiederum Requests für die PostgreSQL Datenbank und das Bitfinex-API verarbeitet. Da anfangs noch nicht klar war, ob für jeden Benutzer eine eigene Orderleiter-Maintainer Instanz erzeugt werden sollte, welche für die Requests an das Bitfinex-API verantwortlich ist, wurde entschieden, dass alle Benutzer auf dieselbe Orderleiter-Maintainer Instanz zugreifen, was zur Folge hatte, dass diese für asynchrone API-Calls umgeschrieben werden musste. Für die Entwicklung sowohl als auch das Deployment in die Produktion wurde Docker verwendet.

Vorgehen

Zu Beginn der Arbeit wurden mit dem Auftraggeber die Anforderungen an die Weboberfläche erarbeitet. In wöchentlichen Meetings wurden diese kontrolliert und offene Fragen besprochen. Die Entwicklung lief agil ab, wobei Änderungen und Verbesserungen schnell umgesetzt werden konnten.

Resultate

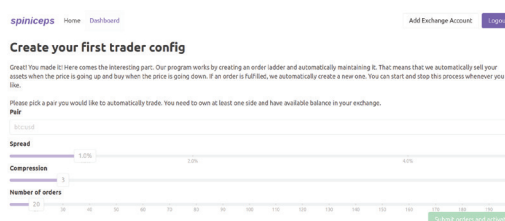
Alle nötigen Anforderungen konnten soweit erfolgreich in einer ersten Version umgesetzt werden. In einem nächsten Schritt wäre es erdenklich, die Applikation durch ausgewählte Nutzer testen zu lassen, welche Feedback geben, damit allfällige Verbesserungsvorschläge später implementiert werden können.



Lorenz Frank Roswin Bischof



Fabio Caggiano



Hier kann der Kunde ein Trading-Pair konfigurieren und den Orderleiter-Maintainer aktivieren.