# IP6: Blockchain Transactionmanager

Projektvereinbarung

Faustina Bruno; Jurij Maïkoff

**Studiengang:**
- iCompetence
- Informatik

**Betreuer:**
- Markus Knecht
- Daniel Kröni

**Auftraggeber:**
Fachhochschule Nordwestschweiz
FHNW Campus Brugg-Windisch
Bahnhofstrasse 6
5210 Windisch

2019-10-01

# Inhaltsverzeichnis

# 1 Aufgabenstellung

Blockchains verfügen über verschiedene Mechanismen um sich gegen Attacken abzusichern. Eine davon ist eine Gebühr auf jeder Transaktion, die sogenannte Gas Fee. Dadruch können Denail of Service (DoS) Attacken, bei denen das Netzwerk mit unzähligen Transaktionen geflutet wird, effizient bekämpft werden. Der Angreifer kann die Attacke nicht aufrecht erhalten, da ihm die finanziellen Mittel ausgehen.

Obwohl dieser Schutzmechanismus auf einer öffentlichen Blockchain sehr effizient und elegant ist, eignet er sich nicht für eine Lernumgebung. Hier sollen Anwender die Möglichkeit haben, Transaktionen ohne anfallende Gebühren ausführen zu können. Dadurch wird jedoch die Blockchain anfällig für DoS Attacken.

Die Projektaufgabe besteht darin, eine Lösung zu finden, bei der die Sicherheit der Blockchain auch ohne eine Transaktionsgebühr gewährleistet werden kann.

## 1.1 Ziele

Das Ziel der Arbeit ist es zuerst eine Konzeptionelle Erarbeitung eines Testnetzwerkes welches:

- nicht permanent ist (Reboot möglich)
- kostenlose Transaktionen ermöglicht
- Anonymität gewährleistet
- Sicherheit gewährleistet

und in einem zweiten Schritt die Umsetzung/Realisierung dieses Netzwerkes.

Um diese Ziele zu erriechen sind folgende Fragestellungen von Bedeutung:

- wie kann die Gebühr für Transaktionen auf null gesetzt und die Sicherheit der Blockchain trotzdem gewährleistet werden
- Unterstützt uPort[1] unsere gewünschten Anfroderungen einer SmartWallet oder müssen wir selber eine SmartWallet programmieren
- Wie kann man Attacken vermeiden (zB algorithmisch: nur eine beschränkte Anzahl Transaktionen pro Monat pro Benutzer möglich)

## 1.2 Risiken

**Tabelle 1.1:** Risiken

| Risiko | Auftreten | Auswirkung | Kategorie | Gegenmassnahme |
|---|---|---|---|---|
| Teammitglied bricht Projekt ab | 1 | 3 | 3 | Gute Kommunikation unter den Teammitgliedern |
| Unterschätzen des Projektumfanges | 2 | 2 | 4 | Sorgfältige Planung und regelmässig Rücksprache mit den Betreuern |
| Ausfall von einem Teammitglied (mehr als 2 Wochen) | 2 | 2 | 4 | Sofortiges Informieren von Betreuern. Planung überarbeiten und Ausfall berücksichtigen |

In der Tabelle 1.1 sind die wichtigsten Risken aufgelistet. In der Spalte Auftreten wird die gschätzte Wahrscheinlich eines Eintreffen des Risikos beschrieben. Die Spalte Auswirkung beschreibt die Schwere beim Eintreffen des Risikos. Bei beiden Spalten ist der Wert 1 das Minimum und der Wert 3 das Maximum. Der Wert in der Spalte Kategorie wird aus der Multiplikation von Auftreten und Auswirkung gebildet. Ein Risiko kann also von 1 bis 9 gewertet werden. Je höher die Kategorie, umso gefährlicher ist ein Risiko.

# 2 Entwicklungsumgebung

In diesem Abschnitt wird die geplante Testumgebung und deren Verwendung beschrieben.

## 2.1 Betriebssystem

Beide Teammitglieder verwenden Windows 10 als Betriebssystem.

## 2.2 Blockchain

Um unser erworbenes Wissen auch testen zu können, wird eine Test-Blockchain aufgesetzt. Zu Beginn bietet die Testumgebung eine Möglichkeit, das Gelernte anzuwenden und so das Verständnis für das Thema zu vertiefen. Später im Projekt wird die Umgebung benötigt um ausgearbeitete Ansätze zu testen und analysieren.

Als Blockchain wird Ethereum[2] verwendet.

## 2.3 Wallet

Für die Verwaltung von Identitäten und Transaktionen auf einer Blockchain werden sogenannte Wallets verwendet. Diese Verwaltung ist auch auf einer Lernumgebung nötig, daher muss geprüft werden, ob vorhandene Wallets, wie zum Beispiel uPort[1], unseren Ansprüchen genügen oder ob diese im Rahmen von diesem Projekt entwickelt werden müssen.

## 2.4 Smart Contracts

Für die Entwicklung von Smart Contracts wird die Sprache Solidity[3] verwendet. Auch hier wird die Testumgebung genutzt, um Gelerntes anweden zu können. Sobald eigene Smart Contracts entwickelt werden, kann die Testumgebung genutzt werden, um diese zu testen.

# 3 Termine

**Tabelle 3.1:** Grober Zeitplan

| Datum | Event |
|---|---|
| 24.09.2019 | Kickoff |
| NOV | Erster Konzept Entwurf und Testnetzwerk |
| 28.11.2019 | Zwischenpräsentation |
| DEZ | erste Konzept Version |
| FEB | Testen von MVP |
| 20.03.2019 | Abgabe Bachelorthesis |
| 13. - 24.04.2019 | Verteidigung |

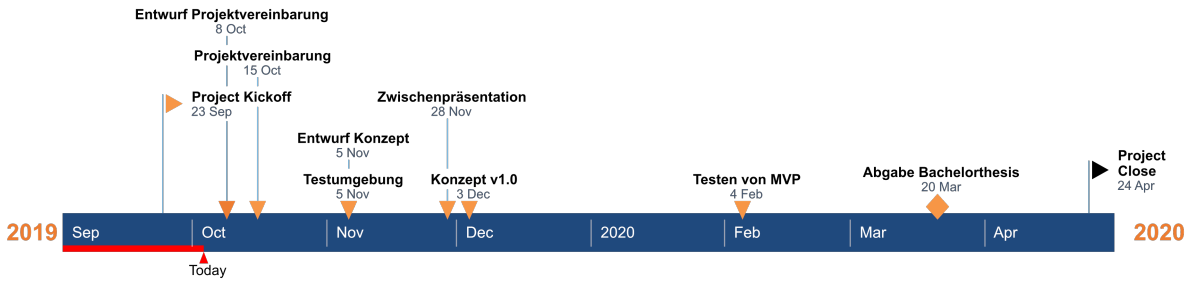


**Abbildung 3.1:** Zeitstrahl

In der Grafik 3.1 sind die bereits bekannten Meilensteine eingetragen. Im Laufe des Projekts wird die Grafik aktualisiert.

# 4 Quellenverzeichnis

[1] uPort, „uPort“, 2019. [Online]. Verfügbar unter: https://www.uport.me/.

[2] Ethereum, „Home | Ethereum“, 2019. [Online]. Verfügbar unter: https://www.ethereum.org/.

[3] Solidity, „Solidity - Solidity 0.5.11 documentation“, 2019. [Online]. Verfügbar unter: https://solidity.readthedocs.io/en/v0.5.11/.