

Aufgabenbeschreibung „Zimmervermietung“

Inhaltsverzeichnis

Gegebenes Szenario	2
Minimum	2
Erweiterung	2
Grafik	2
Aufbau eines Angebots	3
Für ein Mietzimmer	3
Umzusetzende Punkte und Abzuklärendes	4
Smart Contract mit Ethereum	4
Whisper Messages Protokoll für Ethereum	4
web3.js Client bei Ethereum Blockchain	4
REST-Schnittstelle bei Hyperledger Blockchain	4
User Account für UI	4
User Interface	4
Abzuklärendes	4
Transaktion durchführen	4
Mögliche Meilensteine	5
Organisatorisches	5
Kleingruppen	5
Veranstaltungstermine	5
Kurzvorträge	5
Seminararbeit	6
Präsentation	6

Gegebenes Szenario

Minimum

- 1 Zimmer: kann immer wieder vermietet werden
- 1 Mieter: mietet ein Zimmer
- 1 Vermieter: vermietet ein Zimmer

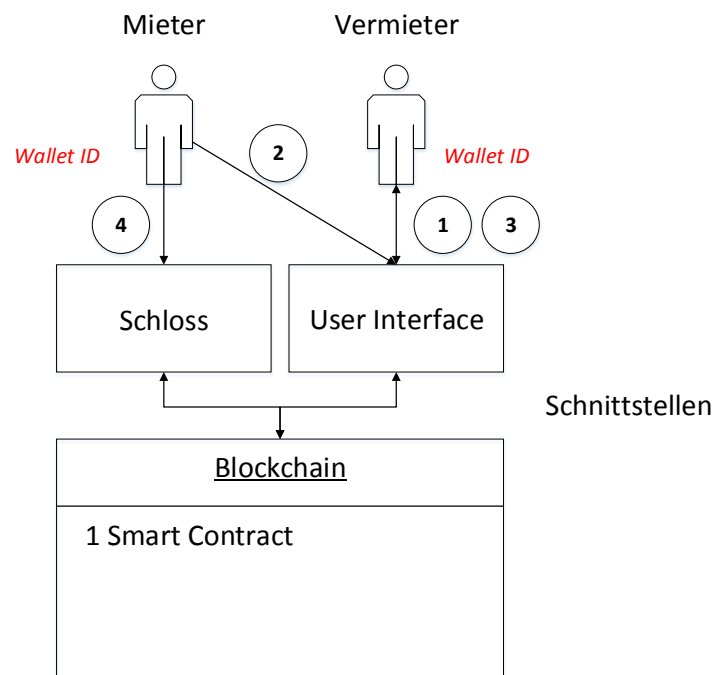
Anmerkungen:

1. Ein Zimmer entspricht einer Tür.
2. Es ist die Möglichkeit gegeben, während des gebuchten Zeitraums das Schloss mehrmals zu öffnen (Umsetzung mit Whisper).
3. Das Schloss soll als Hardware vorhanden sein.

Erweiterung

- Mehrere Zimmer
- Mehrere Mieter
- Mehrere Vermieter

Grafik



- 1 Vermieter legt Angebot über User Interface in Blockchain an (= Transaktion)
- 2 Mieter schaut, welche Angebote es gibt (Only Read) und entscheidet sich ggf.
- 3 Vermieter führt Buchung durch (nach Buchungsanfrage durch den Mieter, welche nicht vom Vermieter abgelehnt werden kann)
- 4 Mieter kann Schloss mit Smartphone öffnen

Anmerkungen:

1. Ein Smart Contract, welcher in einer Datenbank verwaltet wird, um die Daten immer auf den aktuellsten Stand zu halten.
2. In einem Smart Contract in der Blockchain wird immer das gleiche Booking durchgeführt.
3. Wie sieht die Kommunikation zwischen User Interface und Blockchain aus?
4. Ggf. im User Interface auswählen welche Technologie eingesetzt werden soll (Ethereum oder Hyperledger).

Aufbau eines Angebots

Für ein Mietzimmer

Variable	Datentyp
Angebot ID	Long
Tür ID	Long
Mietpreis (pro Nacht)	Long
Gültigkeitszeitraum Inserat (Zeit von /Zeit bis)	Date (Verwendung von UNIX)
Buchungszeitraum Zimmer (Check-in /Check-out)	Date (Verwendung von UNIX)
Adresse des Objekts	Text
Name des Vermieters	String
Mieter Wallet ID → <i>Public Key</i>	ca. 20 zufällige Zeichen
Vermieter Wallet ID → <i>Public Key</i>	ca. 20 zufällige Zeichen
Beschreibung	String
evtl. Zimmerbild	

Anmerkungen:

1. Tür ID ist Zieladresse für Whisper (quasi auch *Public Key*).
2. Gültigkeitszeitraum Inserat wird durch den Vermieter festgelegt (Zeitraum kann auch leer gelassen werden und ist dadurch bis zur Löschung gültig).
3. Buchungszeitraum Zimmer wird durch den Mieter festgelegt.
4. Mieter Wallet ID und Vermieter Wallet ID sind „quasi“ die Benutzernamen.
5. *Public* und *Private Key* werden automatisch generiert und werden für die Transaktionen benötigt.
6. Mieter Wallet ID und Vermieter Wallet ID in JSON-File oder Datenbank (Serverseitig) speichern?

Umzusetzende Punkte und Abzuklärendes

Smart Contract mit Ethereum

- Ein oder mehrere Smart Contracts pro Angebot
- Verwaltung von Public Keys (Ethereum-Seite) und überlegen wie diese in Block geschrieben werden
- Umsetzung mit dem Framework Truffle + Ganache
- Smart Contract ID referenziert immer auf Ursprungsblock
- Welche Aktionen werden innerhalb des Smart Contracts ausgeführt?

Whisper Messages Protokoll für Ethereum

- Skizze bereits vorhanden
- Keine Transaktionen (und dadurch keine Kosten)
- Integriertes Schloss (Hardware) → Zugriff über Blockchain
- Smartphone sendet Token, welches zur Authentifizierung des Mieters durch den Vermieter dient
- Whisper signiert und hinterlegt einen Public Key

web3.js Client bei Ethereum Blockchain

- Umsetzung mit Angular und einem Node
- Anbindung der Daten in der Blockchain allgemein
- Kommunikation zwischen Blockchain und User Interface

REST-Schnittstelle bei Hyperledger Blockchain

- Anbindung der Daten in der Blockchain allgemein
- Kommunikation zwischen Blockchain und User Interface

User Account für UI

- Eigene Verwaltung von Accounts, Geld und Keys
- Plugin oder direkt über Ethereum bzw. Hyperledger Wallet

User Interface

- Schickt Nachrichten an die Tür
- Webinterface (grafische Benutzeroberfläche)
- Frontend Darstellung / wie buche ich überhaupt?

Abzuklärendes

- Bei Whisper für Ethereum Verwendung von Public Key (P2P) oder Wallet ID
- Public und Private Key Verwaltung auf Diversen Geräten
- Vorhandenen Code für die Authentifizierung in unsere Applikation einbinden und evtl. für Hyperledger umschreiben bzw. erweitern

Transaktion durchführen

- Hyperledger: Private Key, Username (Wallet ID? Fingerprint) / E-Mail Adresse
- Ethereum: Key Phrase → Private Key, Wallet IDs / E-Mail Adresse
 - Schnittstelle zum Handy gibt es fertig als App

Mögliche Meilensteine

Ausbauen und genauer definieren (mit Datumangaben)!

Tür: Hardware, Tür geht auf durch Whisper (mit Überprüfung)

Hyperledger: Problem sind die Schnittstellen zu den anderen

1. Architektur festlegen (Skizze)
2. Schloss und User Interface Implementierungen & Implementierung und Integration der Komponenten in der Blockchain
3. Mieter und Vermieter mit Wallets einrichten
4. Testdurchführung des Minimums mit 1 Zimmer, 1 Mieter und 1 Vermieter
5. Evtl. Umsetzung für mehrere Zimmer / Mieter / Vermieter

Organisatorisches

- Pflege von Einzelaufgaben und Einzelnen Projektteilen übernimmt jede Kleingruppe selbst.
- Besprechung immer Dienstag 30-60 Minuten. Danach Arbeiten in Kleingruppen.

Kleingruppen

- Hyperledger: Deniz, Frank
- Tür mit Whisper: Felix, Max, Peter
- Smart Contract: Anna, James, Rene
- Web3 Client: Alex, Daniel, Michi

Veranstaltungstermine

- Dienstag 29.05.2018
- Dienstag 05.06.2018
- Dienstag 12.06.2018
- Dienstag 19.06.2018
- Dienstag 26.06.2018

Kurzvorträge

Team-Mitglied	Thema	Datum
Anna		
Alex		
Daniel	OpenChain	24.04.2018
Deniz	Hyperledger	08.05.2018
Felix		
Frank	Hyperledger	08.05.2018
James	Smart Contracts und Ethereum	29.05.2018
Max		
Michi		
Peter		
Rene		

Seminararbeit

- Umfang maximal 5 Seiten pro Team-Mitglied (Text und Bilder)
- Kapitel mit Namen kennzeichnen
- Anna fügt alles zusammen und kümmert sich um das Layout etc.
- Abgabe (durch Anna) am Freitag den **13.07.2018** per E-Mail als PDF-Datei

Präsentation

- Dienstag **03.07.2018** und Dienstag **10.7.2018** *oder* Montag 02.07. *oder* Donnerstag 05.07. jeweils **ab 17 Uhr**
- Vortrag im Team; Dauer ca. 10 Minuten pro Team-Mitglied
- Umfang: Aspekte der Umsetzung/Vergleiche/andere Möglichkeiten/Probleme etc.