

Projektarbeit

Entwurf, Implementierung und Evaluation einer prototypischen Lösung für Bestell- und Bezahlprozesse auf Blockchain-Basis in mittelständischen Unternehmen

Eingereicht von: Marcel Antzek, Bastian Kühnel,
Felix Leber, Christopher Lemm,
Sönke Rückwardt, Daniel Schick

Betreuung: M.Sc. Tim Reimers

Modul: IT-Management
Modulnummer 1047

Abgabedatum: 23. März 2018

Abstract

Im Verlauf dieser Arbeit wird die Blockchain-Technologie beschrieben. Ziel ist es durch diese Technologie Chancen und Risiken für die IT erfassen und bewerten zu können. Hierzu wird zunächst eine Machbarkeitsstudie und eine Risikoanalyse für ein entsprechendes Tool angefertigt. Anschließend wird die Blockchain-Technologie für einen spezifischen Anwendungskontext in einem Demonstrator (WebApp) umgesetzt. Der Demonstrator wird sich dem Bestell- und Bezahlungsprozess annehmen. Außerdem werden Interviews mit kleinen Unternehmen des Gastronomiesektors durchgeführt, um festzustellen, wie diese die Möglichkeiten der Technologie wahrnehmen und den Demonstrator bewerten. Diese Bewertungen werden als Indizien für weitere Entwicklungsstufen des Demonstrators herangezogen. Die gesammelten Ergebnisse werden schließlich evaluiert und zusammengefasst.

Inhaltsverzeichnis

Tabellenverzeichnis	VII
Listings	VII
Abkürzungsverzeichnis	IX
1. Einleitung	1
1.1. Motivation	1
1.2. Aufbau des Dokumentes	2
2. Grundlagen der Blockchain-Technologie	5
2.1. Kryptographische Hashfunktionen	5
2.2. Dezentralisierte Netzwerkstrukturen	6
2.3. Asymmetrische Kryptographie	7
2.4. Kryptowährungen	7
2.4.1. Bitcoin	8
2.4.2. Ethereum	9
2.4.3. Vergleich der exemplarischen Konzepte	9
3. Methodisches Vorgehen	11
3.1. Methodische Grundlagen der Interviews	11
3.2. Projektmanagement-Werkzeug <i>Trello</i>	12
3.3. Risikoanalyse	14
4. Szenariodefinition	17
4.1. Betriebswirtschaftliche Kontextualisierung	17
4.2. Fallbeispiel: Whoppercoin Burger King	19
4.3. Use Cases im Gastronomie-Kontext	20
4.3.1. Perspektive: Kunde	23
4.3.2. Perspektive: Betreiber	24
5. Implementierungsprotokoll Demonstrator-App	27
5.1. Aufbau	27
5.2. Smart-Contracts	27
5.2.1. Entscheidungen für das User Interface	32
6. Evaluation	35
6.1. Durchführung der Interviews	35
6.2. Risikoanalyse	37

Inhaltsverzeichnis

7. Konklusion	41
7.1. Zusammenfassung	41
7.2. Ausblick	42
Anhang A. Transkript der Interviews	45
Literaturverzeichnis	55

Abbildungsverzeichnis

1.1.	Schematische Darstellung des Blockchain-Konzeptes	2
1.2.	Logos der Blockchain-Anwendungen Ethereum, Bitcoin und Hyperledger	2
2.1.	Netztopologien	6
3.1.	Übersicht des <i>Trello</i> -basierten Kanban-Boards	13
3.2.	Exemplarische Karte des <i>Trello</i> -basierten Kanban-Boards	14
4.1.	Repräsentative Unternehmensumfrage zur Digitalisierung 2017	18
4.2.	Beispiel eines UML-Diagramms für einen Anwendungsfall	21
4.3.	Exemplarisches <i>Slicing</i> eines monolithischen Use Cases	22
5.1.	Ansicht für Besucher	32
5.2.	Ansicht für Mitarbeiter:Bestellungs-Übersicht	33
5.3.	Ansicht für Mitarbeiter: Verifikation durch Etherscan	34
6.1.	SWOT-Analyse der Protolösung	39

Tabellenverzeichnis

5.1.	Untereinheiten der Ethereum-Währung	28
6.1.	Themen für die Experteninterviews	35

Listings

5.1.	Drinks: defining manager	29
5.2.	Drinks: price-functions	30
5.3.	Drinks: order, server and fallback-functions	31

Abkürzungsverzeichnis

ABI	Application Binary Interface
BTC	Bitcoin
DApp	Decentralized Application
ETH	Ethereum
EVM	Ethereum Virtual Machine
GUI	Graphical user interface
HTTP	Hypertext Transfer Protocol
ID	Identifikator
IoT	Internet of Things
IP	Internet Protocol
ISO	International Organization for Standardization
IT	Informations-Technologie
LAN	Local area network
P2P	Peer-to-Peer
SWOT	Strengths, Weaknesses, Opportunities und Threats
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UML	Unified Modeling Language
URL	Uniform Resource Locator
VM	Virtual machine
VoIP	Voice over IP
WAN	Wide area network
WWW	World wide web

1. Einleitung

Diese Arbeit wird eine Abhandlung über die Blockchain beziehungsweise ihrer Anwendungsmöglichkeit in spezifischem Kontext sein. Genauer werden Einsatzgebiete beleuchtet und Chancen die sich für die IT und somit auch die Nutzer daraus ergeben. Es werden aber auch mögliche Risiken aufgezeigt.

1.1. Motivation

Wir leben im 21. Jahrhundert, indem die Bedeutung von Daten immer mehr zunimmt. Eine Vielzahl von Geräten verfügt mittlerweile über einen Speicher irgend einer Form und auch besonders die Anzahl der Geräte, welche vernetzt operieren nimmt drastisch zu. Der daraus resultierende Zuwachs des Wertes von Daten geht soweit, dass in einem Artikel „der Zeit“ sogar soweit gegangen wird davon zu sprechen, dass Daten das Gold des 21. Jahrhunderts seien [1]. Häufig gibt es für solche Daten keinen zentralen IT-Dienst, weshalb diese dezentral beispielsweise mittels P2P-Netzen umgesetzt werden. Was das genau bedeutet und wie so etwas funktioniert wird in Kapitel 2.2 beschrieben. In vielen Fällen ist die Korrektheit dieser Daten von essenzieller Bedeutung für verschiedenste Parteien. Betrachtet man das Beispiel, dass 2 Parteien ein Vertrag in .PDF-Form vorliegen haben. Sagt die eine Datei nun, dass beispielsweise eine 1000 Euro Zahlung fällig wird und die andere sagt es würden 2000 Euro fällig, birgt dies ein deutliches Konfliktpotential. Ohne eine Überprüfungsmöglichkeit ist hier ein Streit vorprogrammiert und eine Lösung lässt sich möglicherweise nur schwer erzielen. Sind diese Daten nun aber mit einem Mechanismus versehen, der sie auf Echtheit und Richtigkeit prüfen kann ist es möglich potentielle Konflikteskalationen zu vermeiden. Die beiden Datensätze könnten einem unabhängigen Gericht vorgelegt werden und anhand dieser Mechanismen könnte Recht gesprochen werden. Der Streit wäre gelöst.

Ein solcher Mechanismus zum Nachweisen der Korrektheit und Integrität könnte beispielsweise durch eine Blockchain realisiert werden. Sehr grob beschrieben ist die Blockchain ein Mechanismus, der vergangene Informations-Fragmente beziehungsweise Versionsdaten in seiner sogenannten Headerinformation referenziert. Neue Informationen werden dabei in sogenannten Blöcken angefügt, wodurch der tatsächliche Informationsgehalt der Blockchain-Struktur stetig wächst. Dieser Mechanismus wird in Abbildung 1.1 veranschaulicht, wobei die entsprechende Version der Komponente repräsentiert. Eine detaillierte Beschreibung der technischen Parameter des Blockchain-Konzeptes erfolgt in Abschnitt 2.

1.2. Aufbau des Dokumentes

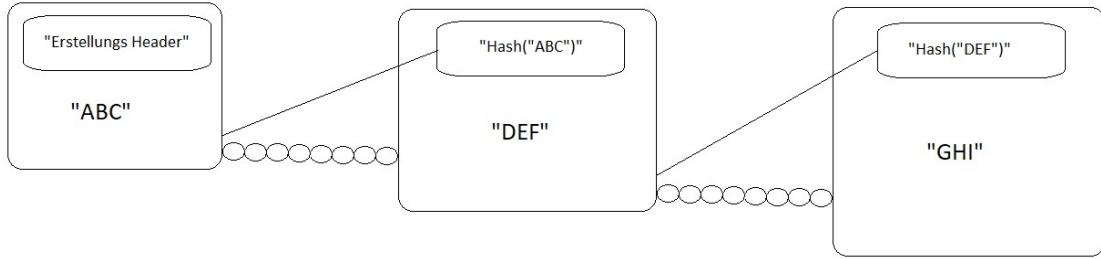


Abbildung 1.1.: Schematische Darstellung des Blockchain-Konzeptes

Die wohl bekanntesten Anwendungsfälle der Blockchain-Technologie sind aktuell die Kryptowährungen. Allen voran ist hier der sogenannten Bitcoin zu nennen. Bitcoin ist eine 2009 unter dem Pseudonym Satoshi Nakamoto, über dessen wahre Identität oder Identitäten bis heute nur spekuliert werden kann, veröffentlichte Kryptowährung. Bitcoin war laut Statista (Stand 2017) die mit deutlichem Abstand dominante Kryptowährung [2]. Als nächstes zu nennen und Bitcoin auf Rang 2 der Kryptowährungen folgend ist Ethereum, welches im Jahr 2013 von Vitalik Buterin vorgestellt wurde und im Jahr 2015 erschien. Ethereum selbst ist eigentlich nicht die Kryptowährung, sondern das System, welches mit der Währung ETH arbeitet. Der Unterschied zwischen Bitcoin und Ethereum ist, dass Ethereum zusätzlich sogenannte „Smart Contracts“ verarbeitet, welche so viel wie „Intelligente Verträge“ bedeuten. Die beiden bisher genannten Beispiele unterliegen im Vergleich zu „echten“ Währungen keiner zentralen Verwaltung wie beispielsweise Banken, sondern ihr Wert bestimmt sich zu 100% aus Angebot und Nachfrage. Weniger bekannt, aber nicht weniger relevant ist das Hyperledger Project. Hierbei handelt es sich nicht um eine Blockchain an sich, sondern eher um eine Ansammlung von Open Source Blockchain Tools. Diese Tools beschäftigen sich damit Daten überprüfbar und sicher zu machen. Auch bietet Hyperledger einige Frameworks zum Entwickeln eigener Blockchain Tools an, um diese weiter zu verbreiten. Die jeweiligen Firmenzeichen der drei referenzierten Projekte sind in Abbildung 1.2 unten übersichtsartig dargestellt.



Abbildung 1.2.: Logos der Blockchain-Anwendungen Ethereum, Bitcoin und Hyperledger

1.2. Aufbau des Dokumentes

In Kapitel 1.1 wurde eine Einführung in das Thema Blockchain gegeben. In Kapitel 2 wird auf die theoretischen Grundlagen eingegangen und jene erläutert. Zunächst wird dort

1.2. Aufbau des Dokumentes

auf Kryptohashes eingegangen. Anschließend wird allgemein P2P (Peer-to-Peer) erläutert und darauf folgend dezentrale Netze erklärt. Danach wird Asymmetrische Kryptographie betrachtet und schließlich mit der genaueren Erläuterung des Begriffs Kryptowährung abgeschlossen. In Kapitel 3 wird das methodische Vorgehen in dieser Arbeit erläutert. Dazu gehören unter anderem die Durchführung von Interviews und einer Machbarkeitsstudie. In Kapitel 4 wird zunächst eine betriebswirtschaftliche Kontextualisierung durchgeführt. Anschließend wird ein Fallbeispiel geliefert und verschiedene Use-Cases aus dem Bereich der Gastronomie aufgezeigt. In Kapitel 5 wird ein entworfener Prototyp vorgestellt sowie für diesen benötigte beziehungsweise durch ihn durchgeführte Smart Contracts erläutert. Die Arbeit schließt mit dem Kapitel 7 einem Fazit.

2. Grundlagen der Blockchain-Technologie

Allgemein lässt sich eine Blockchain als geordnete, rückverkettete Liste von Blöcken, also als eine verteilte Datenstruktur[3], in der digitale Datensätze, Ereignisse oder Transaktionen gespeichert werden, beschreiben. Durch die Verwendung von Kryptohash-Werten und asymmetrischer Verschlüsselung wird die Fälschungssicherheit der Blockchain gewährleistet. Ein dezentralisiertes Peer-to-Peer Netzwerk verteilt die Blockchain auf eine Vielzahl von Knotenpunkten, um die Verfügbarkeit und Ausfallsicherheit zu erhöhen[4].

2.1. Kryptographische Hashfunktionen

Hash- oder Streuwertfunktionen sind Funktionen, die eine beliebig große Eingabemenge auf eine kleinere Ausgabemenge abbilden. Die Länge der Eingabe spielt hierbei keine Rollen, wobei die Länge der Ausgaben immer gleich sein sollte. Hashfunktionen sind deterministisch, das heißt bei gleicher Eingabe muss auch das gleiche Ergebnis erzielt werden. Diese Berechnung der Hash-Werte soll möglichst schnell von statthen gehen. Für einige Anwendungsfälle ist ein zusätzliches Kriterium, dass wenn sich ein Zeichen des Eingabewertes ändert, sich möglichst viele des Stellen des Zielwertes ändern sollen. **[Shannonsche Konfusion/Diffusion, Avalanche-Effekt]** Ein Rückschluss von Hash- auf Eingabewert soll nicht effektiv möglich sein, was man als „Einwegfunktion“ bezeichnet [5]. Die bisher genannten Eigenschaften sollten sowohl auf Hashfunktionen als auch Kryptologische Hashfunktionen zutreffen. Bedingungen, die für eine kryptologische Hashfunktion aber für eine normale Hashfunktion nicht zwingend erfüllt sein müssen, sind die schwache und starke Kollisionsresistenz. Schwache Kollisionsresistenz bedeutet, dass es nicht effektiv möglich sein darf zu einem gegebenen Eingabewert einen weiteren Wert zu finden, sodass deren beide Ergebnishashes gleich sind. Starke Kollisionsresistenz bedeutet, dass es nicht effektiv möglich sein darf überhaupt ein solches Paar aus der gesamten möglichen Eingabemenge zu finden, deren Ergebnishashes gleich sind. Einige Kryptologische Hashfunktionen haben zusätzlich die Möglichkeit einen zusätzlichen Schlüssel für die Funktion anzugeben, sodass beispielsweise zwei Werte mit unterschiedlichem Schlüssel auch unterschiedliche Hashes erzeugen. Diese Eigenschaft ist allerdings kein zwingendes Kriterium für eine Kryptologische Hashfunktion. Die 4 Kernanforderungen für Kryptologische Hashes werden in der folgenden Übersicht dargestellt:

1. Effiziente / Schnelle Berechnung
2. Einwegfunktion
3. Schwache Kollisionsresistenz
4. Starke Kolissionsresistenz

Aufgrund der immer größer werdenden Rechenleistungen von Computern und beispielsweise dafür eingesetzten Großrechnern oder Computer-Cluster gibt es weite Diskussionen darüber, welche (Kryptologischen) Hashfunktionen tatsächlich (noch) sicher sind. Die am weitesten verbreiteten Kryptologischen Hashfunktionen sind die SHA-“X“ und MD“X“ Reihen, wobei „X“ für den jeweiligen Versionsname steht [6]. Auch bei diesen Hash-Familien wird mittlerweile von der Verwendung alter Versionen aufgrund mangelnder Sicherheit abgeraten. Bei den aktuellen Vertretern dieser Hashes geht man aber von ausreichender Sicherheit aus.

2.2. Dezentralisierte Netzwerkstrukturen

Ein Grundprinzip der Blockchain ist das Peer-to-Peer Netzwerk. Teilnehmer des Netzwerkes werden Knoten oder Peer genannt. In Abbildung 2.1 ist ein zentralisiertes Netzwerk zu sehen. Alle Knoten sind über eine zentrale Komponente mit einander verbunden. Der gesamte Netzwerkverkehr liegt auf Seiten der zentralen Verbindungskomponente. Des Weiteren ist eine Kommunikation im Netz bei Ausfall der zentralen Komponente nicht mehr gewährleistet. Die Blockchain verwendet ein verteiltes bzw. vermaschtes Netzwerk. Dieses ist in Abbildung 2.1 dargestellt. In diesem Netzwerk existiert keine zentrale Komponente. Viel mehr entstehen durch Verbindungen zu mehreren Knoten Redundanzen im Netzwerk. Solange der gegenüberliegende Knoten aktiv ist, kann mit ihm kommuniziert werden, ohne dass die Daten auf einem Server oder einer anderen zentralen Stelle zwischengespeichert werden müssen. Falls einer der Knoten ausfällt, gibt es noch weitere Knoten, auf die zurückgegriffen werden kann. In der Regel werden mehrere Verbindungen aufrecht erhalten, um die Redundanz zu gewährleisten.

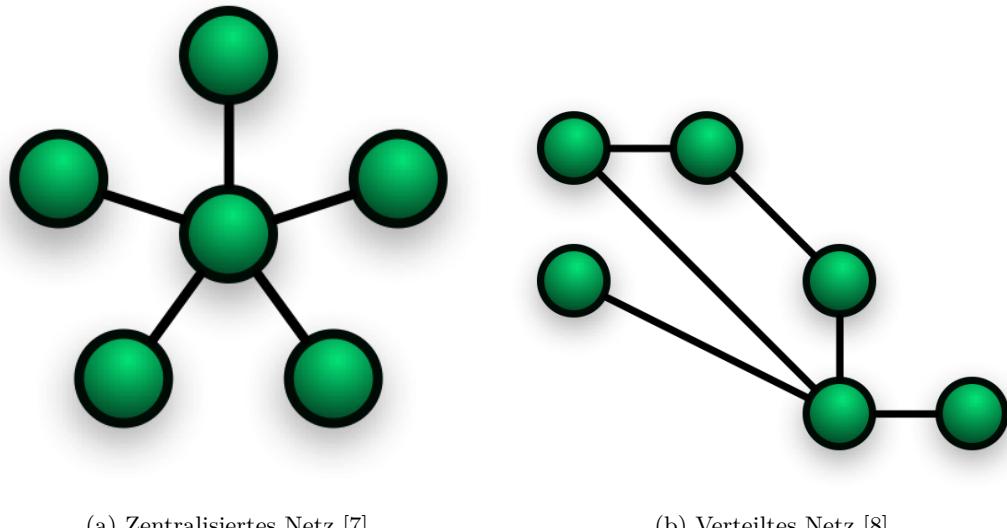


Abbildung 2.1.: Netztopologien

2.3. Asymmetrische Kryptographie

Die Blockchain wird über ein Peer-to-Peer Netzwerk auf die einzelnen Konten verteilt. Transaktionen innerhalb des Netzwerkes werden als neuer Zustand in der Blockchain gespeichert. Jeder Knoten des Netzwerkes hält eine Kopie der Blockchain. Der Begriff der Dezentralisierung bekommt bei der Blockchain eine besondere Bedeutung. Die Dezentralisierung des Netzwerkes sorgt nicht nur für Ausfallsicherheit, sondern auch für Unabhängigkeit. So ist es praktisch unmöglich für einen einzelen die Informationen, die in der Blockchain gespeichert sind zu verändern oder zu zensieren.

2.3. Asymmetrische Kryptographie

Um an einem auf Blockchain basierenden System teilzunehmen, braucht es eine Zugangssoftware. Die Zugangssoftware wird als Wallet bezeichnet und besteht aus einem öffentlichen und privaten Schlüssel. Die aus der asymmetrischen Verschlüsselung stammende Technik wird zur Signierung und Adressierung der kryptografischen Währung genutzt [9]. Zu den bekanntesten asymmetrischen Verschlüsselungsverfahren zählen RSA und Diffie-Hellman [?]. Der öffentliche Schlüssel dient als Adresse der Wallet, vergleichbar mit der IBAN-Nummer eines Kontos. Außerdem kann mit Hilfe des öffentlichen Schlüssels, der Kontostand einer Wallet eingesehen werden. Jedoch dient der öffentliche Schlüssel als Pseudonym, so dass es nicht möglich ist, aus ihm die Identität des Besitzers zu ermitteln. Dieser öffentliche Schlüssel wird bekannt gegeben. Für Transaktionen benötigt man den eigenen privaten Schlüssel, welcher als Pin der eigenen Wallet fungiert [?]. Der Absender kombiniert die Nachricht mit seinem privaten Schlüssel und sendet die signierte Nachricht an die sich aus dem öffentlichen Schlüssel ergebende Adresse des Empfängers. Mit dem öffentlichen Schlüssel des Absenders kann der Empfänger die signierte Nachricht prüfen und somit die Authentizität der Nachricht verifizieren [10]. Der Absender der Nachricht kann nicht leugnen, diese signiert zu haben [11]. Durch die assymmetrische Verschlüsselung kann die Nachricht nicht unbemerkt verändert werden, wodurch deren Integrität gewährleistet ist [9].

2.4. Kryptowährungen

Kryptografische Währungen sind eine Art digitales Geld. Im Internet sind diese vielseitig einsetzbar. Zu den größten kryptografischen Währungen zählen Bitcoin und Ethereum. Mit ihnen können online Einkäufe getätigter oder weltweit innerhalb von kürzester Zeit Währungsbeträge versendet werden. Außerdem werden sie wie die üblichen Währungen auf dem Währungsmarkt gehandelt und Investoren nutzen sie als Geldanlage genutzt. Der wesentliche Unterschied zwischen gewöhnlichem Geld und Kryptowährungen ist die Regulierung sowie die Entstehung des Geldes. Normales Geld wird von einer Zentralbank herausgegeben. Im Vergleich dazu entstehen kryptografische Währungen durch Blockchain-Prozesse.

Kryptowährungen existieren bis auf wenige Ausnahmen nur in der Blockchain. Zu diesen Ausnahmen zählt IOTA. Über eine Wallet erhält der Besitzer Zugriff auf seine Kryptowährungsanlagen. Da Digitalwährungen nur in der Blockchain existieren, gibt es sie auch

2.4. Kryptowährungen

nicht in Form von Bargeld im klassischen Sinne. Von einigen Währungen, beispielsweise Bitcoin, können jedoch auch echte Münzen erworben werden, die den geheimen Code eines einzelnen Bitcoins beinhalten. Allerdings ist hierbei besondere Vorsicht geboten, denn solange der Code noch nicht in die Wallet des Erwerbers übertragen wurde, ist er anfällig für Angriffe und kann sogar von dem Verkäufer der Münze „zurückgeklaut??“ werden, falls dieser den Code noch kennt. [12]

2.4.1. Bitcoin

Bitcoin war die erste Kryptowährung und ist maßgeblich dafür verantwortlich, dass der Begriff Blockchain heute so bekannt ist [13]. Das Bitcoinsystem ist vollständig und gut in einem Wiki [14] dokumentiert und wird in vielen Arbeiten behandelt oder als Beispiel genutzt. Das Bitcoin-Konzept wurde ertsmals 2008 unter dem Pseudonym Satoshi Nakamoto in einem White Paper vorgeschlagen [15]. Das Einheitenzeichen für Bitcoin ist BTC, welches auch oft als Abkürzung verwendet wird.

Für die Generierung von Bitcoins müssen Knoten des Peer-to-Peer Netzwerkes Lösungen zu einem bestimmten, schwer lösbarer mathematischen Problem finden. In den ersten vier Jahren seit Bestehen des Bitcoin-Netzwerkes wurden 10.500.000 Bitcoin geschaffen. Aller vier Jahre wird dieser Betrag halbiert, sodass sich über die Zeit, die Gesamtzahl an Bitcoins 21 Millionen annähern wird. Der letzte Block, welcher eine Münze generiert, wird mit dem jetzigen System etwa im Jahr 2140 erreicht werden. Die derzeitige Größe der Bitcoin-Blockchain beträgt 159 Gigabyte (Stand März 2018) [16].

Im Bitcoinsystem wird der Konsensmechanismus Proof-of-Work verwendet. Das Ziel aller Konsensmechanismen ist es, einen Konsens zwischen gegenseitig nicht vertrauenswürdigen Teilnehmern ohne vertrauenswürdigen Dritten zu bilden [3]. Der Grundgedanke von Konsensmechanismen ist es, dass kein Teilnehmer allein den aktuellen Zustand des Netzwerkes oder eines Teils davon bestimmen kann, es gleichzeitig aber jedem Teilnehmer potenziell möglich ist den Zustand des Netzwerkes zu verändern. Der jeweilige Konsensmechanismus gibt die Bedingungen vor, die ein Teilnehmer erfüllen muss, um den Zustand des Netzwerkes verändern zu dürfen. Aus dieser Bedingung wird eine Wahrscheinlichkeit abgeleitet, mit der der Teilnehmer den Zustand tatsächlich verändern wird [13]. Im Kontext von Blockchain-Netzwerken besteht die Veränderung darin, einen Block anzuhängen. Oft wird auch von generieren, erzeugen oder minen eines Blockes gesprochen. Die Bedingung muss von einem einzelnen Teilnehmer nur schwer zu erbringen, von allen anderen Teilnehmern aber leicht überprüfbar sein. Dabei bezieht sich schwer oft auf hohen Rechenaufwand oder eine geringe Wahrscheinlichkeit.

Bei Proof-of-Work ist die Bedingung der Nachweis einer gewissen Rechenleistung. Dieser Mechanismus wird von Nakamoto mit *one-vote-per-cpu* beschrieben [17, 15]. In Bitcoin ist dafür in jedem Block eine Zahl enthalten, die so gewählt oder besser gefunden werden muss, dass der Hashwert des Blocks kleiner als ein vom Netzwerk vorgegebener Wert ist. Für jeden Teilnehmer ist der zu berechnende Block unterschiedlich, da die erste Transaktion jedes Blockes eine Transaktion *aus dem Nichts* an den Teilnehmer selbst ist und dort der individuelle öffentliche Schlüssel hinterlegt ist. Dadurch und durch die niedrige Wahrscheinlichkeit die richtige Zahl zu erraten, kann nicht vorhergesagt werden, welcher

2.4. Kryptowährungen

Teilnehmer den nächsten gültigen Block generiert [17]. Proof-of-Work kann theoretisch einen Teilnehmer mit einer Rechenleistung von < 50% der Gesamtrechenleistung des Netzwerkes kompensieren. Praktisch wird das Netzwerk aber schon ab 25% instabil [18].

2.4.2. Ethereum

Das Ethereum-System baut auf dem Prinzip der Blockchain auf und dient zur Handhabung von kryptografischen Währungen. Dabei wird Ether als interne Währung im Ethereum-System verwendet. Im Zusammenhang mit Ethereum sind auch Smart Contracts zu nennen. Dabei handelt es sich um Programme, welche automatisch ausgeführt werden, sobald eine festgelegte Summe Ether überwiesen wurde. Nach der Überweisung startet automatisch die im Vertrag festgelegte Dienstleistung. Smart Contracts werden in der für Ethereum entwickelten Programmiersprache Solidity geschrieben. Des Weiteren können dezentralisierte Applikationen auf der Blockchain ausgeführt werden, diese werden durch Smart Contracts beschrieben.

Ethereum wurde durch die Publikationen *Ethereum: A Next Generation Smart Contract and Decentralized Application Platform* (2013) [19] und *Ethereum Yellow Paper* (2014) [20] beschrieben. Zu den Entwicklern und Mitbegründern des Ethereum-Projekts gehören Vitalik Buterin, Gavin Wood und Jeffrey Wilcke. Der Betrieb des Ethereum-Netzwerkes startete Juli 2015 [21].

Als Konsensmechanismus wird Proof-of-Work verwendet, jedoch ist im Entwicklungsplan von Ethereum vorgesehen, dass der Mechanismus auf Proof-of-Stake verändert wird [22]. Proof-of-Work ist sehr rechenintensiv. Proof-of-Stake als Alternative, ist ein deutlich weniger rechenaufwendiger Mechanismus. Die Bedingung ist der Besitz von Anteilen von Token im Netzwerk. Je mehr Anteile ein Teilnehmer besitzt, desto wahrscheinlicher generiert er einen Block [13].

2.4.3. Vergleich der exemplarischen Konzepte

Sowohl Bitcoin als auch Ethereum nutzen die Blockchain als Grundlage ihres Netzwerkes. Bei Bitcoin handelt es sich außerdem um eine Währung, Ethereum nutzt dabei Ether als Währung im eigenen System. Bitcoin wurde im Jahr 2009 vorgestellt. Ethereum erstmals im Jahr 2013, allerdings handelt es sich bei Ethereum auch um ein Projekt, welches seit dem ständig fortentwickelt wurde. Erkennbar ist es unter anderem, dass der Konsensmechanismus bei Ethereum im Laufe der Entwicklung von Proof-of-Work zu Proof-of-Stake wird. Bitcoin nutzt seit seiner Vorstellung exklusiv Proof-of-Work als Konsensmechanismus.

Ein wesentlicher Unterschied ist die Verwendung von Smart Contracts und dezentralisierten Applikationen bei Ethereum. Dadurch wird die Nutzung des Ethereum-Netzwerkes flexibler im Vergleich zum Bitcoin-Netzwerk. Durch Smart Contracts entstehen viele neue Anwendungsfelder für die Blockchain-Technologie. Erwähnenswert sind Lizenzvergabe, Zahlungsabwicklungen oder Vertragsabschlüsse. In den kommenden Jahren werden weitere Geschäftsfelder Smart Contracts nutzen. Im Zusammenhang mit Smart Contracts ist der Begriff Token zu nennen. Durch den Ethereum Token Standard, auch ERC20 genannt,

2.4. Kryptowährungen

ergeben sich für Entwickler weitere Möglichkeiten Smart Contracts auf der Ethereum-Blockchain zu nutzen. Durch die ständige Weiterentwicklung der Ethereum-Technologie ergeben sich deutlich mehr Anwendungsmöglichkeiten als sie Bitcoin bietet.

3. Methodisches Vorgehen

Nach Nennung und Erläuterung wichtiger technischer sowie theoretischer Grundlagen folgen in diesem Kapitel die methodischen Grundlagen, mit der diese Arbeit erstellt wird. Hierbei wird im ersten Abschnitt auf Interviews eingegangen, welche im weiteren Verlauf geführt werden, um aus Kundensicht den ideal Prototyp einer Ethereum-basierten App herzustellen. Der zweite Abschnitt behandelt das Thema *Trello*, welches genutzt wird, um innerhalb eines Teams die Übersicht über das Projekt zu wahren. Die Grundlagen über Machbarkeitsstudien sowie Risikoanalysen bilden das Ende des Kapitels.

3.1. Methodische Grundlagen der Interviews

Um aus Herstellersicht zu erfahren, inwiefern sich ein Produkt für ein Kunden eignet und welche Schwerpunkte bei der Entwicklung gelegt werden müssen, ist es nötig Anforderungen und Wünsche zu ermitteln. Dies kann auf mehreren Wegen geschehen, beispielsweise in Form von printbasierten Medien. Der Nachteil hierbei ist allerdings, dass vorgedruckte Fragen oftmals zu ungenau auf kundenspezifische Anliegen eingehen. Eine aufwändiger, allerdings deutlich persönlichere und daher kundenähnere Form der Befragung ist das Interview. Hierbei stehen Hersteller sowie Kunde im direkten Kontakt. Somit ist es möglich, innerhalb eines Gesprächs detailliert auf konkrete Wünsche und Vorstellungen des Kunden einzugehen und etwaige Problematiken zu diskutieren. Ein weiterer Vorteil ist die Beschreibung der Arbeitsabläufe seitens des Herstellers, was zu mehr Verständnis durch den Kunden führt und so die Arbeit in der Regel für alle Parteien erleichtert. Ein Nachteil des Interviews ist der erhöhte Zeit- und Personenaufwand. In diesem Abschnitt werden die Rahmenbedingungen für das Interview gelegt, welches im weiteren Verlauf geführt wird, um die Anforderungen an einen möglichen Prototypen festzustellen.

Die Vorbereitung für ein Interview betreffen beide Seiten, wobei primär der Hersteller betroffen ist. Dieser hat die Aufgabe, spezifische Fragen zu stellen und möglichst viele Wünsche des Kunden zu erfassen und konkrete Anforderungen daraus abzuleiten. Je detaillierter die daraus resultierenden Antworten sind, desto spezifischer kann der Hersteller im weiteren Verlauf sein Produkt auf die Wünsche des Kunden anpassen. Der Hersteller hat die Aufgabe, das richtige Personal für ein Interview bereitzustellen. Dieses Team besteht im Idealfall aus Experten, welche sich bereits mit der Thematik, in diesem Falle Ethereum und App Entwicklung, auskennen und den Kunden im Laufe des Interviews beraten können. Aus Kundensicht ist es nötig, im Voraus Anforderungen und konkrete Vorstellungen beschreiben zu können. Während des Interviews ist es beiden Seiten möglich, den Gesprächsverlauf dynamisch zu beeinflussen und so bisher nicht erkannte Probleme anzusprechen. Das Interview sollte gut dokumentiert werden, um eine spätere Auswertung für den Hersteller zu ermöglichen. Eine abschließende Zusammenfassung

3.2. Projektmanagement-Werkzeug *Trello*

der wichtigsten Punkte, insbesondere noch ungeklärter Fragen, helfen sowohl Kunde als auch Dienstanbieter bei der Entwicklung. Es ist daher möglich, neben dem initialen Interview auch weitere persönliche Gespräche zu führen, um das Produkt bestmöglich an den Kunden anzupassen. Um die wichtigsten Aspekte beim Einführen einer Software im Gastronomiebereich zu etablieren, werden Experteninterviews durchgeführt. Es werden Gastronomien im umliegenden Bereich interviewt. Daber wird eine semigeleitete Interviewform gewählt, um möglichst individuelle Informationen, wie zum Beispiel Wissensstand über Blockchain oder dem tatsächlichen Einsatz des Szenarios in dem Betrieb. Diese Informationen werden anschließend ausgewertet, um eine Ausblick für weitere Untersuchungen zu schaffen. Auf Herstellerseite ist es von Nutzen, besonders für kleinere Projekte wie die in dieser Arbeit erstellte Ethereum App, mithilfe von spezieller Software Anregungen des Kunden festzuhalten und den aktuellen Fortschritt des Produkts festzuhalten. Ein Beispiel für diese Software ist *Trello*, welche im nächsten Abschnitt näher erläutert wird.

3.2. Projektmanagement-Werkzeug *Trello*

Aufgrund des limitierten Projektumfangs sowie der Notwendigkeit zur Einarbeitung von dynamischen Änderungsanfragen wurde im Rahmen der Projekt- und Dokumententwicklung ein agiler Projektmanagement-Prozess realisiert, welcher intern auf einer Kombination der klassischen Vorgehensmodelle *Scrum* und *Kanban* basiert [11]. Die Projektbearbeitung zeichnet sich bei Verwendung dieser agilen Modelle insbesondere durch ein iteratives Vorgehen in enger Abstimmung mit dem Auftraggeber aus, sodass die phasenweise produzierten Produkt-Artefakte kontinuierlich verfeinert werden können. Die Realisierung von zentralen Teilkomponenten des finalen Produkts wird dabei innerhalb sogenannter *Sprints* aggregiert, welche eine zeitlich limitiert Phase mit dediziertem Implementierungsschwerpunkt repräsentieren. Zur Koordination des konkreten Entwicklungsprozesses wurde bei Projektinitialisierung die Verwendung eines zentralen Kanban-Boards beschlossen, welches primär als tabellarisches Visualisierungselement der projektinternen Teilaufgaben fungieren sollte. Auf diese Weise können die anfallenden Aufgaben explizit ausgewählten Projektmitgliedern zugewiesen werden, welche die jeweiligen Teilaufgaben auf Basis des internen Fortschritts in verschiedene Entwicklungs-Stufen klassifizieren können. Aufgrund der limitierten Ressourcenlage sowie des hohen Zusatzaufwandes, den der lokale Betrieb einer dedizierten Software-Lösung wie *Atlassian Jira* oder *OpenProject* induziert, fiel die Entscheidung auf die Nutzung des kostenfreien Web-Anbieters *Trello*. Innerhalb der Trello-Umgebung wird ein dynamisch rekonfigurierbares Kanban-Board bereitgestellt, dessen Inhalte interaktiv in der Browser-Umgebung administriert werden können. Die Struktur kann hierbei individuell an die individuellen Projektanforderungen angepasst werden; aufgrund der Einfachheit in der Bedienung sowie der Möglichkeit zur kollaborativen Nutzung mit weiteren Mitgliedern hebt sich das Tool hierbei von vergleichbaren Anbietern ab [11]. Trello fasst die Projektmitglieder zu einem Team zusammen, innerhalb dessen unter Verwendung von grafischen Visualisierungshilfen das Anlegen und die Bearbeitung von Teilaufgaben (in der Trello-Terminologie

3.2. Projektmanagement-Werkzeug *Trello*

als *Karten* bezeichnet) koordiniert werden kann. Die Aufgaben können dabei durch individuelle Markierungen gezielt an einzelne Teammitglieder delegiert werden, sodass die Zuständigkeiten innerhalb des Projekts stets klar geklärt sind. Weiterhin können im Kontext der Aufgaben-Spezifikation integrierte Zusatzfunktionalitäten wie Fristen und Checklisten genutzt werden, sodass beispielsweise eine schrittweise Realisierung einzelner Teilaufgaben ermöglicht wird.

Strukturell wurde das Kanban-Board für die Anwendungs- und Dokumententwicklung in fünf Spalten separiert, wobei die drei elementaren Entwicklungs-Stufen der Aufgaben durch ein initiales Kommunikationselement und eine Übersicht der aktuellen Sprint-Ziele eingeschlossen werden. Diese grundlegende Konfiguration der Trello-Umgebung nach Abschluss der ersten Initialisierung ist exemplarisch in Abbildung 3.1 unten dargestellt.

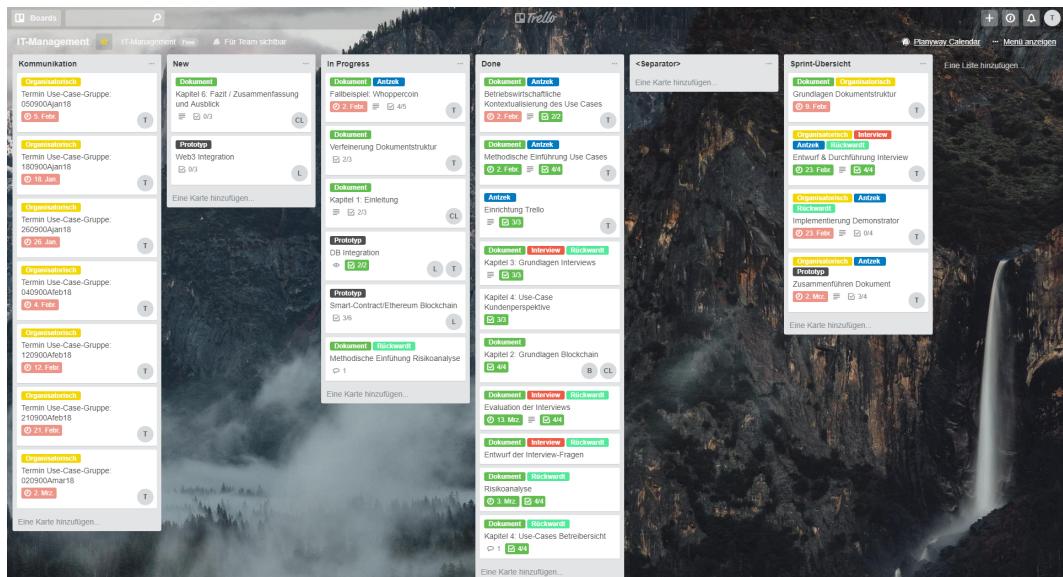


Abbildung 3.1.: Übersicht des *Trello*-basierten Kanban-Boards

Die drei zentral angeordneten Spalten *New*, *In Progress* und *Done* repräsentieren die angesprochenen Fortschrittsstufen der individuellen Teilaufgaben. Initial werden neue Aufgaben innerhalb der Spalte *New* deklariert und einem Team-Mitglied zugewiesen, welches vor Beginn der Bearbeitung die internen Parameter und Schnittstellen exakt spezifiziert. Sobald die Arbeit an einer Teilaufgabe begonnen wird, wird das korrespondierende Karten-Element für alle sichtbar in die Spalte *In Progress* verschoben. Auf diese Weise verfügen alle Teammitglieder über eine aktuelle Übersicht des Bearbeitungsstatus; dies ermöglicht bei Konflikten oder partiellen Überlappungen eine vereinfachte Koordination der beteiligten Akteure. Nach Abschluss der Bearbeitung erfolgt die Verschiebung der Karte in die Spalte *Done*, wobei anhand der internen Checklisten des Karten-Elements die Vollständigkeit der Bearbeitung validiert werden kann. Die Resultate werden final durch einen Vertreter der Gruppenleitung überprüft, sodass eine konfliktfreie Integration der

3.3. Risikoanalyse

Teilresultate in den globalen Bearbeitungsstand garantiert werden kann. Die Akzeptanz einer Teilaufgabe durch die Gruppenführung wurde projektintern durch Aktivieren der Frist-Checkbox signalisiert, welche die Karte für den weiteren Projektverlauf als inaktiv markiert (vergleiche Abbildung 3.2 unten).

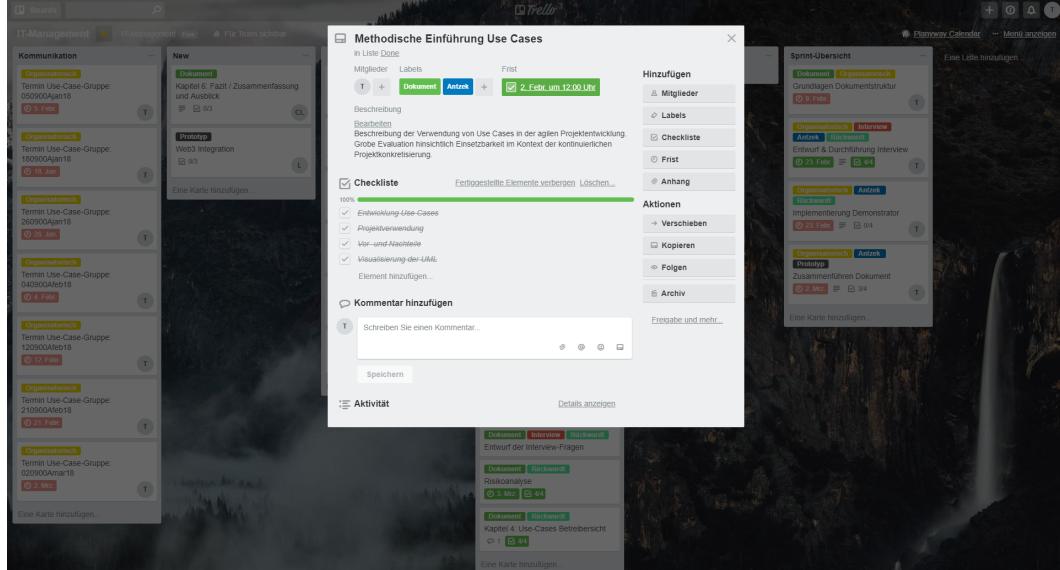


Abbildung 3.2.: Exemplarische Karte des *Trello*-basierten Kanban-Boards

Durch die Verwendung einer internen Projektmanagement-Anwendung konnten im Projektverlauf Überlappungen und Kollisionen in der Aufgabenbearbeitung weitestgehend vermieden werden. Die Wahl eines agilen Konzeptes ermöglichte dabei die Anpassung an individuelle Projektparameter, ohne jedoch übermäßigen Verwaltungsaufwand zu induzieren.

3.3. Risikoanalyse

Nach der Betrachtung der UseCases erfolgt eine kurze Risikoanalyse des Produktes [23]. Im Rahmen des Projektmanagements bedeuten Risiken in erster Linie Unsicherheiten, die sich negativ auf den Projektverlauf auswirken können. Definiert man das Risiko rein mathematisch, handelt es sich um einen Zustand, der eintreffen kann oder nicht. In Betrachtung des Projektmanagements sind Risiken jedoch reale und virtuelle Ereignisse, die einen realen Schaden am Projekt hervorrufen können. Der Eintritt eines Risikos zieht also immer negative Auswirkungen mit sich. Dazu zählen folgende drei Faktoren, die alle drei betroffen sein können, aber nicht zwingend müssen: Zeit, Kosten und Qualität. Die Risikoanalyse im Rahmen dieses Projektes hat zum Ziel, Risiken im fortlaufenden Projekt zu erkennen, zu analysieren und die Wahrscheinlichkeit des Eintreffens der Risiken mit den daraus resultierenden Folgen zu ermitteln. Als weiteres Ziel schafft die Risikoanalyse die Möglichkeit einen Entscheidungsprozess zu optimieren und zu objektivieren. Der erste Schritt zur Durchführung einer Risikoanalyse im Rahmen des Projektmanagements

3.3. Risikoanalyse

besteht darin Risiken zu erkennen. Bei der Ermittlung der Risiken ist zu beachten, dass zwischen verschiedenen Risiken (externe Risiken, interne Risiken, planerische Risiken, kaufmännische-, fachliche- und Umfeld-Risiken) unterschieden wird. Die nachfolgenden Schritte dienen der Risikoübersicht, der Befragung von Projektbeteiligten, dem Studium der Projektunterlagen und der Analyse im laufenden Betrieb. Das Wegfallen und Entstehen von Risiken begleitet das gesamte Projekt über die gesamte Laufzeit. Dies macht eine ergänzende Analyse der Risiken, ihrer Wahrscheinlichkeiten und Folgen unabdingbar.

4. Szenariodefinition

Auf Basis der innerhalb des vorhergehenden Kapitels formulierten technischen sowie methodischen Kontextualisierung werden im Rahmen dieses Abschnittes multiperspektivisch exemplarische Nutzungsszenarien der Blockchain-Technologie diskutiert und final hinsichtlich ihrer Realisierbarkeit in produktiven Umgebungen des Gastronomie-Sektors evaluiert. Aufbauend auf einer initialen Definition der betriebswirtschaftlichen Terminologie werden dabei aus den Erfahrungen der *Whoppercoin*-Implementierung elementare Use-Cases zur Entwicklung neuer Anwendungskonzepte für den gastronomischen Bereich extrahiert.

4.1. Betriebswirtschaftliche Kontextualisierung

Innerhalb der letzten Jahre ist sowohl für den privaten als auch für den industriellen Sektor eine kontinuierliche Intensivierung der Digitalisierungsanstrengungen zu konstatieren. Insbesondere im gewerblichen Kontext sind kontinuierliche Optimierungen der digitalen Infrastruktur zu einer wesentlichen Voraussetzung für wirtschaftlichen Erfolg und internationale Konkurrenzfähigkeit geworden; unterstützt durch die permanente technologische Entwicklung sind innerhalb produktiver Unternehmensumgebungen automatisierte Datenverarbeitungs-Prozesse sowie digitalisierte Kommunikationsstrukturen heute weitestgehend omnipräsent. Gemäß der im Jahre 2015 publizierten Studie *d!conomy - Die nächste Stufe der Digitalisierung* des Bundesverbands Informationswirtschaft, Telekommunikation und neue Medien e. V. (Bitkom) bewertet die Majorität der Unternehmen in Deutschland mit 86% (im industriellen Sektor sogar 93%) Zustimmung die durch Digitalisierung induzierten Transformationsprozesse als positiv [11]; auf Basis dieser Stimmungslage prädisziert das Marktforschungsinstitut *Kantar Taylor Nelson Sofres* (Kantar TNS) 2017 die in Abbildung 4.1 visualisierte, signifikante Inkrementierung des innerdeutschen Digitalisierungsgrades aller Branchenzweige bis zum Jahr 2022 [11].

Primär wird die sukzessive Einführung einer digitalisierten Infrastruktur für die Unternehmen dabei durch die potentielle Effizienzmaximierung in klassischen Sektoren wie Produktion, Logistik und Ein- und Verkauf motiviert [11]; sekundäre Aspekte werden durch die Optimierung interner Prozesssteuerungs- und Controlling-Elemente sowie die Verbesserung der Kundeninteraktion repräsentiert, deren Priorisierung in Abhängigkeit vom individuellen Unternehmensschwerpunkt markant divergieren kann [11].

Die moderne Interpretation des Terminus *Digitalisierung* erweitert den ursprünglichen Optimierungsgedanken um einen weiteren Aspekt, welcher die IT in der Rolle eines Prozess-*Enablers* darstellt. Auf Basis der durch Informationstechnologie offerierten Funktionalität können dabei fundamental neue Geschäftsmodelle, Produkte und Dienstleistungen generiert werden; IT-basierte Innovationen fungieren somit nicht länger nur

4.1. Betriebswirtschaftliche Kontextualisierung

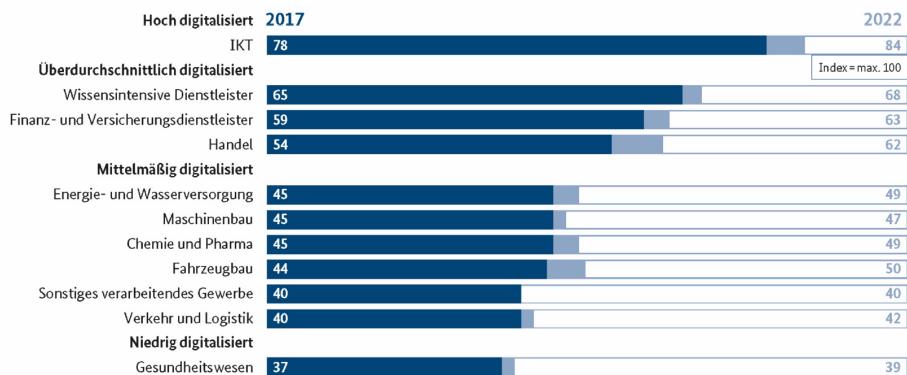


Abbildung 4.1.: Repräsentative Unternehmensumfrage zur Digitalisierung 2017 (nach [11])

zur Optimierung bereits bestehender Prozessstrukturen, sondern können in ihrer Rolle als unmittelbare Voraussetzung (*Enabler*) die radikale Reorganisation von Prozessen und Wertschöpfungsketten induzieren [11]. Konzeptionell können die identifizierten IT-Innovationsansätze nach dem Kriterium des initialen Innovationsimpulses kategorisiert werden, wobei grobgranal zwischen technologieinitiiertem Transformationsdruck (*technology push*) und dem externen Marktsog (*market pull*) differenziert wird. Im Kontext des *technology push* wird das Innovationselement durch Resultate technologischer Entwicklung repräsentiert, welche potentiell die Generierung bisher nicht existenter Marktsegmente induzieren und somit hohes Ertragspotenzial bieten. Die konträre Strategie des *market pull* realisiert hingegen die Extraktion von potentiellen Kundenbedürfnissen aus der dynamischen Analyse bereits existierender Marktanteile, auf deren Grundlage technische Optimierungsansätze für Produkte, Dienste und Anwendungen abgeleitet werden [11,12]. Im konkreten Unternehmenskontext ist unter Berücksichtigung der individuellen Schwerpunkte jedoch meist eine Form der hybriden Implementierung zu präferieren, um die individuellen Defizite der Konzepte durch selektive Rekombination ausgewählter Faktoren zu eliminieren. Die strikt isolierte Realisierung einer Innovationsstrategie kann mitunter in der Gefährdung der Marktposition resultieren; so besteht bei *technology push* das Risiko zur Entwicklung eines Produktes ohne Markt, während im Kontext des *market pull* der Fokus auf kontinuierliche Minimaloptimierungen des Ursprungprodukts (auch als *facelift* bezeichnet) die Realisierung tatsächlicher Innovationen obstruieren kann [11]. Ausgehend von dem individuellem Leistungs- und Wachstumspotenzial eines Unternehmens ist dabei die Konzeption einer adaptiven Selektionsstrategie elementar wichtig, um unter Berücksichtigung von dynamischen Marktumgebungen und potentiell temporären Tendenzen erfolgversprechende Produkt- und Dienstleistungsparameter zu definieren.

Ein kritisches Element in der unternehmensinternen Digitalisierungsstrategie wird durch die als *Business-IT-Alignment* bezeichnete, strategische Synchronisation von produktiver Unternehmenskomponente und den korrespondierenden IT-Abteilungen repräsentiert. Nur die absolute Kongruenz in der Zielspezifikation von Unternehmensleitung und

4.2. Fallbeispiel: Whoppercoin Burger King

IT-Administration erlaubt die Generierung der reziproken Synergieeffekte, welche den Transfer der IT-Innovationen in den produktiven Kontext ermöglichen und somit final in der Stabilisierung und weiteren Expansion des individuellen Marktanteils resultieren [11].

4.2. Fallbeispiel: Whoppercoin Burger King

Wie bereits im Rahmen der vorangehenden Ausführungen dargelegt, kann die Entwicklung neuer *Enabler*-Technologien signifikante Änderungen in verschiedenen Geschäftsbereiche induzieren. Aufgrund der weitestgehend universellen Einsetzbarkeit als dezentralisiertes Transaktionssystem werden aktuell das in Abschnitt 2.4.1 präsentierte Blockchain-Konzept sowie die korrespondierenden Erweiterungen als potentielle Innovationselemente für den Betrieb zahlreicher Branchenbereiche gehandelt. Der Entstehungsprozess der Blockchain-Technologie wird dabei durch eine Kombination der Entwicklungsstrategien *business pull* und *technology push* repräsentiert. Als eines der initialen Motive zur Realisierung der Blockchain-Implementierungen wird mitunter der marktseitige Bedarf nach einem digitalen, dezentralisierten Währungssystem, sodass im Entstehungskontext von Bitcoin und vergleichbaren Blockchain-Implementierungen eine grundlegende Korrelation mit dem Konzept des *business pull* konstatiert werden kann. Im Rahmen der kontinuierlichen Forschungsintensivierung innerhalb dieses Themenkontextes konnten jedoch sukzessive auch innovative Einsatzszenarien durch die entwickelten Technologien induziert werden, welche insbesondere unter Verwendung generischer Plattformen wie Ethereum zunehmend diversifizierte Funktionalität offerieren können.

Die hierdurch generierte, mediale Präsenz der Blockchain-Thematik visualisiert das aktuelle Expansionspotential dieses Marktsegments, welches von Analysten der Banken- und Finanzbranche zunehmend als Investitionsfeld der Zukunft propagiert wird [11,11]. Die Partizipation in der Blockchain-Forschung sowie die Adaption der entwickelten Technologielösungen in produktiven Unternehmensteilen ermöglichen den Unternehmen dabei die Manifestation oder Expansion ihrer Positionierung innerhalb des korrespondierenden Marktsegmentes, wobei die reinen Optimierungsaspekte der offerierten Funktionalität durch das marketing-technische Instrumentalisierungspotential der weitestgehend positiven Blockchain-Reputation unterstützt werden. Exemplarisch seien an dieser Stelle die signifikanten Kursanstiege der Unternehmen *Kodak* und *Long Island Ice Tea* in Reaktion auf deren partielle Blockchain-Einführung zu Jahresbeginn 2018 referenziert [11,12].

Analog hierzu kündigte im August 2017 auch die US-amerikanische Schnellrestaurantkette *Burger King* für die nahe Zukunft die Akzeptanz von Bitcoin-Zahlungen sowie die Implementierung einer eigenen Cryptowährung, intern als *Whoppercoin* bezeichnet, an. In der aktuellen Konfiguration repräsentiert die vorerst auf russische Filialen limitierte Technologie die Testphase einer dedizierten Kundenbindungsmaßnahme, welche die Auszahlung der digitalen Währung äquivalent zu bekannten Treuepunkte-Konzepten realisiert. Technologisch basiert die Einführung dabei auf der Instrumentalisierung einer bereits existierenden Blockchain-Infrastruktur, welche durch die Handelsplattform des externen Drittanbieters *Waves Go*. offeriert wird [11]. Durch diese Form der Kooperation kann die Generierung potentieller Synergieeffekte zwischen den Unternehmen forciert werden;

4.3. Use Cases im Gastronomie-Kontext

unter Minimierung des individuellen Risikos für die beteiligten Instanzen bei gleichzeitiger Partizipation an potentiellen Gewinnen kann eine vorsichtige Evaluation dieses Marktsegments bei weitestgehender Gefährdungsminimierung für die individuelle Kerngeschäfte realisiert werden. Dieses Vorgehen illustriert die ubiquitäre Ambivalenz aus grundlegender Skepsis gegenüber dieser potentiell revolutionären Datenverwaltungs-Technologie und dem individuellen Innovationswillen der Unternehmen bei der Einführung dieser Technologien. Wesentliche Parameter des vorgestellten *Whoppercoin*-Konzeptes werden dabei jedoch von Währungsspezialisten und Sicherheitsanalysten als kritisch rezensiert; insbesondere die namentliche Bindung der digitalen Währung an ein dediziertes Unternehmen könnte die Universalität möglicher Einsatzszenarien limitieren, da die Akzeptanz der *Whoppercoin*-Währung durch direkte Konkurrenten wie *McDonald's* als unwahrscheinlich einzustufen ist [11]. Weiterhin kann eine enge Namenskorrelation zwischen Unternehmen und Währung final in kritischen Imageschäden für den Betreiber resultieren, wenn Kontributionen des entsprechenden Währungsderivats zu Transaktionen im Kontext strafrechtlich relevanter Tätigkeiten identifiziert und publiziert werden [11]. Aktuell präsentiert sich Burger King mit dem vorgestellten Konzept als branchenübergreifender Vorreiter, wobei das Nachziehen weiterer Unternehmen zeitnah erwartet werden kann.

4.3. Use Cases im Gastronomie-Kontext

Aufbauend auf den theoretischen Erkenntnissen der vorherigen Abschnitte soll im Rahmen dieses Dokumentes eine exemplarische Demonstrator-Implementierung für die Blockchain-Anwendung im gastronomischen Kontext konzipiert und realisiert werden. Hierzu werden initial die zentralen Anwendungsfälle aus Perspektive der beteiligten Akteure identifiziert, wobei auch potentielle Vor- und Nachteile der Blockchain-Technologie illustriert werden. Die manuelle Spezifikation von *Use Cases* (deutsch: Anwendungsfälle), deren konzeptionelle Grundlage 1987 durch Ivar Jacobson publiziert wurde, ermöglicht die Modellierung und Dokumentation der systeminhärenten Funktionalität bereits existierender oder geplanter Produktkomponenten, indem von konkreten Implementierungsparametern abstrahiert mögliche Interaktionsszenarien zwischen Akteuren und dem Zielsystem definiert werden. Die Akteure können hierbei sowohl durch reale Personen als auch durch abstrakte Rollen oder externe Systeme repräsentiert werden, welche die zur Erreichung eines präzisen spezifizierten Ziels erforderlichen Interaktionsmuster mit dem Zielsystem simulieren. Die konkreten Attribute des korrespondierenden Use Cases illustrieren dabei neben der Aktor-Komponente auch die extern erkennbaren Reaktionen des Systems, um final mögliche Konnektionen zwischen Aktorverhalten und systeminternen Abläufen identifizieren zu können. Abläufe werden in diesem Kontext als Folge von individuellen Aktionen definiert, welche final in Erfolg oder Fehlschlag des angestrebten Systemprozesses resultieren; hierbei ist zusätzlich die hierarchische Komposition einzelner Ablaufbeschreibungen möglich, um die funktionelle Systembeschreibung sukzessive in konkrete Implementierungsparameter zu transformieren. Da eine Visualisierung von komplexen Systemprozessen in Form der oftmals grafisch illustrierten Aktionsfolgen die Kommunikation im Rahmen einer

4.3. Use Cases im Gastronomie-Kontext

Projektentwicklung signifikant verbessern kann, wird das Konzept der Use Cases häufig im Bereich der Produktentwicklung und Kundenabstimmung eingesetzt [11].

Die primäre Funktion von Use Cases wird durch die initiale Konkretisierung der im Rahmen eines Projektes zu realisierenden Produktfunktionalität repräsentiert, deren detaillierte Beschreibung die Basis für die Erstellung eines stringenten Entwicklungskonzeptes darstellt. Um spätere Restrukturierungen dieses Konzeptes und damit verbundene Verzögerungen und Kosten zu vermeiden, ist eine möglichst exhaustive Spezifikation aller potentiellen Nutzungsszenarien zu realisieren; insbesondere ist hierbei auch die explizite Dokumentation potentieller Funktionsabbrüche und Bedienungsfehler (sogenannter *Missuse Cases*) erforderlich, um für diese kritischen Fälle ein geeignetes Systemverhalten konzipieren zu können. Die dedizierte Analyse der geplanten Interaktionsprozesse resultiert zusätzlich häufig in der Identifikation weiterer Produktanforderungen, sodass auf Basis einer extensiven Use-Case-Analyse die weitestgehend umfassende Systembetrachtung im Rahmen der initialen Projektplanungs-Phase garantiert werden kann [11]. Der Begriff der Use Cases ist hierbei grundsätzlich generisch und aus diesem Grund nicht exklusiv an ein dediziertes Visualisierungsverfahren gebunden. In produktiven Umgebungen wird jedoch häufig eine standardisierte Diagramm-Form unter Verwendung der *Unified Modelling Language* (UML) realisiert, welche explizit Methoden zur Illustration von System-Funktionalitäten aus Aktorsicht, möglichen Korrelationen zwischen Anwendungsfällen sowie den Beziehung des Systems zur Umwelt formalisiert [11]. Zur Demonstration wesentlicher Attribute der UML-Notation visualisiert Abbildung 4.2 unten exemplarisch die zentralen Prozesse eines Online-Shops (durch die Ellipsen-Elemente repräsentiert) sowie die korrespondierenden Aktor-Instanzen, deren Aktivität durch stilisierte Personenskizzen illustriert wird. Zur Abstraktion von konkreten Implementierungsparametern wird die Betrachtung hierbei auf eine stark simplifizierte Perspektive limitiert, deren Subaspekte im Rahmen des Projektfortschrittes sukzessive konkretisiert werden können.

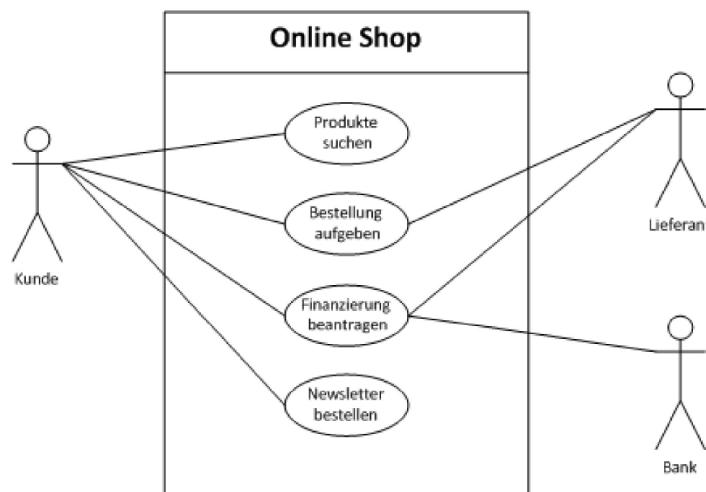


Abbildung 4.2.: Beispiel eines UML-Diagramms für einen Anwendungsfall (nach [11])

Konzept Use Case 2.0

Insbesondere im Kontext agiler Projektentwicklungs-Konzepte ist die Anzahl der potentiellen Use Cases jedoch häufig zu umfangreich, um innerhalb klassischer Projekt-Teilintervalle von wenigen Wochen (im Scrum-Kontext als *Sprints* bezeichnet) aus den Projektbeschreibungen extrahiert und final implementiert zu werden. Zur Lösung dieser Problematik publizierten Ivar Jacobson, Ian Spence und Kurt Bittner im Dezember 2011 das Konzept der *Use Cases 2.0* [11]. Der monolithische Ansatz zur exhaustiven Use-Case-Generierung wird hierbei durch eine skalierbare, agile Technik zur sukzessiven Konkretisierung von Anforderungen substituiert, welche adaptiv an die Verfahrensparameter der inkrementellen System- und Produktentwicklung angepasst werden kann. Die konzeptionelle Grundlage zur agile Projektplanung mit Use Cases liefert dabei das *Slicing*, welches die Zerlegung eines Use Cases in kleinere Teileinheiten beschreibt; die hierbei extrahierten Use-Case-Fragmente werden dabei so angepasst, dass ihre wesentlichen Bestandteile innerhalb eines Projektintervalls realisiert werden können [11].

Die Segmentierung eines Use Cases wird dabei durch iterative Isolation kohärenter Interaktionsfolgen zwischen Aktor und Systeminstanz realisiert, welche final in der Extraktion autarker, stark konkretisierter Anwendungsfälle resultiert. Diese repräsentieren dabei häufig verschiedene, interne Ablauffolgen des analysierten Use Cases; der Grundgedanke der selektiven Pfadauswahl zur *Slicing*-Realisierung ist in Abbildung 4.3 unten dargestellt.

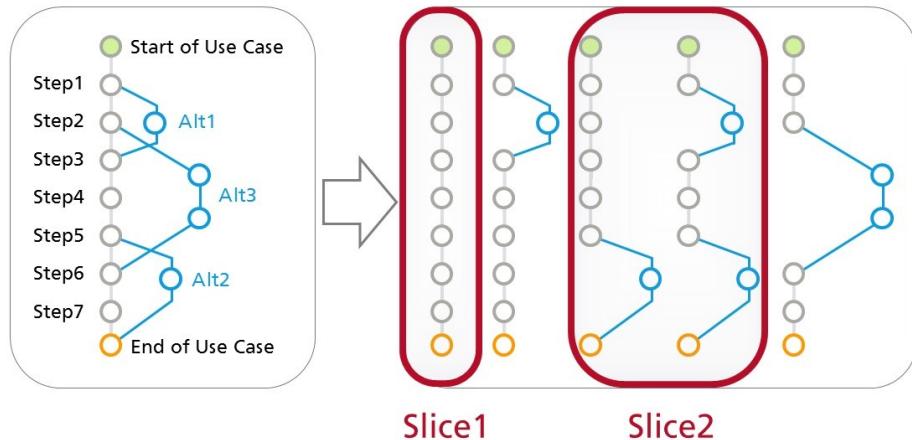


Abbildung 4.3.: Exemplarisches *Slicing* eines monolithischen Use Cases [11]

Zusätzlich wird der Realisierungsaufwand individueller Teilespekte der Anforderungen häufig unter Verwendung eines internen Punktesystems quantifiziert, um auf Basis einer initialen Aufwandsabschätzung die selektive Priorisierung ausgewählter Projektanteile zu ermöglichen. Auf diese Weise kann insbesondere die im agilen Projektkontext elementare Kommunikation mit den Auftraggebern verbessert werden, da Produktartefakte mit wesentlichen Funktionselementen zeitnah generiert und in ihrer Korrektheit (beispielsweise durch manuelle Kontrollen des jeweiligen Auftraggebers) verifiziert werden können [11].

4.3.1. Perspektive: Kunde

In diesem Abschnitt werden die use-cases behandelt, welche aus Kundensicht aufkommen. Anhand eines Beispielszenarios, in dem ein Kunde Getränke bestellt und bezahlt, werden die einzelnen use-cases benannt und detailliert erläutert. Auch Sonderfälle, wie etwa die Stornierung eines Getränktes werden beachtet.

Der durchschnittliche Kunde wird, nachdem er die Bar betreten hat, an einem Tisch platznehmen und dort die Cocktailkarte anschauen. Typischerweise sollten hier die Preise direkt neben den Cocktails stehen. Die Bestellkarte selbst ist online verfügbar, auf der Website der Bar. Durch Berühren des gewünschten Cocktails wird dieser zu einer Bestelliste hinzugefügt. Diese Prozedur wiederholt der Kunde so lange, bis die gewünschte Menge sowie Art an Cocktails auf der Bestelliste steht. Der Gesamtpreis sollte deutlich erkennbar angezeigt werden. Ein Button mit dem Text Bestellen signalisiert das Ende der Bestellung und initiiert den tatsächlichen Einkauf. Das Format der Website muss an Mobiltelefone angepasst sein, da Kunden in der Regel mit diesen bestellen werden. Wird die Bestellung getätig, so werden die benötigten Daten an die Bar übermittelt, sodass ein Mitarbeiter die entsprechenden Cocktails zubereiten kann. Der Kunde bekommt eine visuelle Bestätigung auf der Website, dass seine Bestellung eingegangen ist und bearbeitet wird. Auch eine Zahlungsbestätigung muss angezeigt werden. Die Bestellung selbst kann ebenfalls erneut aufgerufen werden, um für spätere Änderungen verfügbar zu sein. Die Zahlung selbst verläuft per Blockchain im Hintergrund und ist voll automatisiert. Hat der Kunde nicht genug Geld in seinem Wallet, so wird die Bestellung abgebrochen und ein entsprechender Hinweis wird angezeigt. Möchte ein Kunde, nachdem er eine Bestellung aufgegeben hat, die Anzahl oder Art der Cocktails ändern, so muss er die gespeicherte Bestellung auf der Website aufrufen und durch entsprechende Interaktion die Anzahl oder Art der Cocktails verändern. Die Differenz des Betrags wird daraufhin entweder an die Bar überwiesen oder an den Kunden zurückerstattet. Eine entsprechende Bestätigung wird dem Barpersonal sowie dem Kunden angezeigt. Eine Änderung soll nur einmalig möglich sein und, sofern die Bestellung verändert wurde, als Notiz an die Bestellung angeheftet werden. Werden die Cocktails an den Kundentisch geliefert, so wird die Bestellung als abgeschlossen markiert und ist nicht mehr veränderbar. Ist ein Kunde mit der Bestellung unzufrieden, etwa wenn ein Cocktail nicht angenommen und daher auch nicht konsumiert wird, so muss es eine Möglichkeit der Rückerstattung geben. Dies geschieht durch einen speziellen Button, welcher den Wunsch der Rückerstattung an die Website der Bar sendet. Dieser wird, nach Prüfung durch das Personal, entweder gewährt oder gelöscht. Nach Konsum der Cocktails soll es dem Kunden möglich sein, Trinkgeld an den Kellner zu übermitteln. Dies wird, wie bereits mit zahlreichen Optionen davor, mit einem entsprechenden Button auf der Website signalisiert und ist nicht an eine bestimmte Bestellung gekoppelt. Ebenfalls wie bei einem normalen Einkauf geschieht diese Überweisung vollautomatisch und ist nicht reversibel. Eine Bestätigung der Trinkgeldüberweisung wird, nachdem die Transaktion erfolgreich abgeschlossen wurde, unterhalb des Trinkgeldbuttons angezeigt.

4.3.2. Perspektive: Betreiber

In diesem Abschnitt werden die UseCases behandelt, die aus Betreibersicht durch das Tool aufkommen. Zusätzlich werden die Unterschiede zu herkömmlichen Betrieben aufgezeigt, wie sie vielfach auftreten.

Sobald sich der Kunde an einen Tisch gesetzt hat, betrachtet er die Speisekarte, welche manchmal erst vom Wirt oder Kellner gebracht wird. Sobald sich der Kunde für ein Getränk oder eine Speise entschieden hat, tritt der Betreiber erneut an den Tisch und nimmt die Bestellung auf. Erst danach kann der Betreiber mit dem Zubereiten der Speisen oder dem Mixen der Getränke beginnen. Dies ist ein zeitaufwändiger Prozess. Mithilfe der Anwendung lassen sich Bestellung aus Betreibersicht automatisch verarbeiten. Während der Kunde seine Tischnummer angibt und die Bestellung abgibt, werden die Informationen direkt an den Betreiber gesendet. Der Wirt sieht dann, welche Getränke zu welchem Tisch geliefert werden müssen. Dies vermindert die Zeit der Bestellungsaufnahme erheblich und der Wirt hat mehr Zeit, sich um seine Kernkompetenzen zu kümmern, dem Mixen von Getränken. Zusätzlich kann der Kunde auch angeben, dass er die Beratung eines Wirtes sucht oder einen Spezialwunsch hat, sodass der Betreiber auf individuelle Wünsche des Kunden eingehen kann und den Kontakt aufrecht erhält. Damit bleibt der Prozess, trotz der Strukturierten Blockchain mit starren Produkten und Preisen, ein sehr dynamischer Vorgang, der bei Bedarf angepasst werden kann.

Die Buchhaltung ist in Gastronomiebetrieben sehr komplex, da verschiedenste Rechnungen miteinander verbunden werden müssen. In kleineren Betrieben wird die Abrechnung größtenteils auf Papier erstellt. In mittelgroßen bis großen Betrieben, dient eine Software zur Bearbeitung von Rechnungen. Mithilfe der Blockchain Technologie erfolgt die Abrechnung binnen kürzester Zeit. Direkt nach der Bestellung, wird der Kunde gebeten den zu entrichtenden Betrag zu zahlen. Während dem Kunden nur der endgültige Betrag gezeigt wird, wird intern eine aufgeschlüsselte Rechnung gespeichert. So können genau Materialkosten, Personal, Steuern und sonstige in der Rechnung enthaltenden Beträge angezeigt werden. Dies erleichtert die Abrechnung erheblich. Durch die in der Blockchain gespeicherten Informationen können die Rechnungen auch nach langer Zeit noch nachvollzogen und begründet werden. Auch ist es möglich, Trinkgelder in den Beträgen anzugeben und diese bei der Abrechnung zu berücksichtigen. Diese Lösung bietet auch kleineren Betrieben die Möglichkeit, ihre Abrechnung digital und einfach zu speichern und dann weiterzuverarbeiten.

Die Webanwendung ermöglicht außerdem die Anpassung der Speisekarte. Normalerweise müssen alle Speisekarten bei geänderten Preisen neu ausgedruckt werden. Bei Sonderangeboten wird beispielsweise ein zusätzliches Blatt in die Speisekarte gelegt oder eine eigene Tageskarte erstellt. Dieser Prozess ist sehr kostspielig und auch zeitintensiv. Fehler in der Speisekarte können nur mit viel Aufwand wieder korrigiert werden.

Um beispielsweise neue Produkte der Speisekarte hinzuzufügen oder Preise anzupassen ist eine Veränderbarkeit der Speisekarte in der Webanwendung erforderlich. Dadurch ist es möglich, verschiedene Sonderangebote oder neue Produkte ohne Zeitverzögerung in das System einzupflegen und den Kunden direkt die neue Speisekarte zur Verfügung zu stellen. Die Kunden müssen dabei keine neue Seite Aufrufen, da sich die Webseite

4.3. Use Cases im Gastronomie-Kontext

automatisch aktualisiert und die geänderten Preise und Produkte in Echtzeit anzeigt. Die Anwendung lässt sich auch mit herkömmlichen Speisekarten kombinieren, sodass ein autarker Betrieb möglich ist.

Durch den großen Bekanntheitsgrad der Blockchain in den vergangenen Jahren dient das Benutzen dieser Technologie im eigenen Betrieb als gutes Werbemittel [34]. Viele Nutzer sehnen sich außerdem nach Anwendungen für die Blockchain [33]. Sollte eine Gastronomieunternehmen nun bekannt geben, dass es auch eine Möglichkeit zu Bestellung und Bezahlung über die Blockchain ermöglichen, so wird es durch das Alleinstellungsmerkmal neue Kunden gewinnen. Manche Kunden werden allein aufgrund der Technologie und nicht den eigentlichen Produkten das Geschäft betreten und die Anwendung testen. Dies wiederum ermöglicht ein Feedback von Nutzern, die sich gut in dem Bereich Blockchain auskennen, was die Anwendung verbessert und Fehler behebt.

Alle diese UseCases lassen sich problemlos auch mit dem herkömmlichen Betrieb der Gastronomie verbinden. So sind Kunden nicht auf die Blockchain-Technologie angewiesen. Dies erleichtert Personen, die sich nicht mit dem Thema beschäftigen wollen, einen normalen Besuch, ohne mit der Blockchain in Berührung zu kommen.

5. Implementierungsprotokoll Demonstrator-App

5.1. Aufbau

Für den Prototyp eignet sich besonders die Ethereum-Blockchain, da diese die Verwendung von Smart-Contracts auf der Ethereum-Virtual-Machiene (EVM) zulässt und gleichzeitig über eine eigene Währung verfügt. So kann das Senden von Transaktionen mit einer Logik verknüpft und automatisiert werden. Die Berechnungen der Smart-Contracts werden innerhalb einer Virtuellen-Maschine bei den Minern ausgeführt. Als Ausgleich bezahlt der Absender einer Transaktion einen kleinen Betrag. Es kann angenommen werden, dass beim Bezahlen einer Gebühr von ca. 15 Cent (Stand Januar 2018) die Transaktion innerhalb einer Minute von der Ethereum-Blockchain verarbeitet wird. Diese Gebühr ist Abhängig von der Anzahl der Speicher- und Rechenoperationen. Deshalb sollte der Zustand einer Variable auf der Blockchain nur dann verändert werden, wenn dies auch wirklich nötig ist.

Im folgenden Kapitel wird die Implementierung einer DApp beschrieben, also einer Decentralized Application. Decentraliczed Applications bieten eine Schnittstelle zwischen Blockchain und Anwender. Nicht immer ist die Interaktion mit Smart-Contracts für den Nutzer durch ein Wallet intentional oder übersichtlich. Es ist möglich, dass Datentypen falsch verwendet werden und der Nutzer im schlimmsten Fall finanziellen Schaden nimmt. Deshalb bieten DApps eine grafische Benutzeroberfläche, die Beispielsweise auf einem Webserver zur Verfügung gestellt wird. So können auch unerfahrene Nutzer ohne technisches Vorwissen mit der DApp interagieren. Durch Wallets wie MetaMask kann eine Verbindung zwischen dem Ethereum-Account und der DApp hergestellt werden.

Zur Entwicklung des Smart-Contracts wird bei der Entwicklung des Prototypen die Programmiersprache Solidity verwendet. Die Syntax ist von der Javascript-Notation abgeleitet. Beim Kompilieren eines oder mehrerer Solidity-Files entsteht Bytecode, welcher auf der EVM ausgeführt werden kann. Die Schnittstelle zwischen Ethereum und Javascript wird durch die Web3-Bibliothek gewährleistet. Diese wird verwendet, um die Verbindung zwischen Frontend und den auf der Blockchain gespeicherten Daten herzustellen.

5.2. Smart-Contracts

Das folgende Kapitel beschäftigt sich mit der Entwicklung des Smart-Contracts. Dieses Kapitel stellt die Grundlage der Implementierung dar. Mit der verwendeten Programmiersprache Solidity können sogenannte Contracts, also Verträge, definiert werden. Diese

5.2. Smart-Contracts

werden entweder einzeln oder kaskadiert auf der Blockchain ausgeführt. Deshalb wird Solidity auch als Vertrags-Orientiert betitelt [27].

Die Ausführung des Programmcodes wird durch die Einheit Gas vergütet [28]. Das verbrauchte Gas pro Rechen- oder Speicheroperation wird mit dem angegebenen GasPreis multipliziert und muss mit dem Befehl zum Ausführen des Smart-Contracts an die Miner entrichtet werden. Smart-Contracts werden einmal initial an die Blockchains gesandt und der Konstruktor wird ausgeführt [29]. Sobald der Smart-Contract auf der Blockchain ist, können andere Teilnehmer über die sogenannte Application Binary Interface (ABI) auf den Smart-Contract zugreifen und dessen Funktionen ausführen.

Der Smart-Contract selbst ist auf die EVM limitiert. Zwar gibt es globale Einheiten, wie die Zeit in Sekunden, welche seit dem ersten Block vergangen ist messen, jedoch besitzt die EVM kein Zugriff auf externe Speichermedien [27]. Zur Abbildung monetärer Vorgänge ist mit Solidity die Umrechnung in die Untereinheiten von Ethereum wie zum Beispiel in Wei möglich. Dies ist notwendig, wenn wir beispielsweise einen Betrag an den Smart-Contract schicken möchten. Jeder Smart-Contract hat einen eigenen Public-Key und kann über Funktionen, die mit dem Schlüsselbegriff `payable` markiert wurden, die Bezahlung in ETH empfangen. Dieser Betrag wird grundsätzlich in Wei gesendet. Eine Umrechnung ist in der Tabelle 5.1 zu sehen. Diese Aufteilung ist notwendig, um Nachkommastellen von Ethereum-Beträgen entsprechend darstellen zu können, da in der aktuellen Version Fließkommazahlen nicht unterstützt werden [?].

Tabelle 5.1.: Untereinheiten der Ethereum-Währung

Multiplizierer	Name
10^0	Wei
10^{12}	Szabo
10^{15}	Finney
10^{18}	Ether

Der Smart-Contract der DApp soll möglichst die Bestellung aufnehmen können, den entsprechenden Betrag errechnen und den Nutzer dann die Möglichkeit geben, den Betrag zu bezahlen. Der Contract muss jedoch in seiner Funktionalität schlicht gehalten werden, da, wie bereits erwähnt, Rechenleistung sehr teuer werden kann. Es muss also versucht werden, die Übertragungskosten im Verhältnis zu den Kosten des Services möglichst klein zu halten.

Zusätzlich sollte beachtet werden, dass der Besitzer des Smart-Contracts besondere Privilegien erhält. In der Regel wird der Initiator des Smart-Contracts jedoch nicht zwangsläufig der Besitzer des Restaurants sein oder der Besitzer des Restaurants wechselt, weshalb eine Funktion zum Übertragen der Privilegien nötig ist. Diese Funktionalität wird in der `transferPriv`-Funktion gewährleistet (siehe Listing 5.1). Diese Privilegien können es beispielsweise ermöglichen, die Ausschüttung des Gewinnes auszulösen oder den Preis einzelner Produkte zu ändern und neue hinzuzufügen.

5.2. Smart-Contracts

Listing 5.1: Drinks: defining manager

```

1  pragma solidity ^0.4.17;
2
3  /**
4   * @title Drinks, the one and only Bar-DApp
5   * @dev Version of the Drinks-Contract for "IT-Management"
6   */
7  contract Drinks {
8
9    address public manager;
10   mapping(uint => mapping(uint => uint)) public orders;
11   mapping(uint => uint) public prices;
12
13   event EmptyPay(address _sener, uint _amount);
14
15  /**
16   * @dev Set msg.sender as manager
17   * @dev Set prices for drinks in ether
18   */
19   function Drinks() public {
20     manager = msg.sender;
21     prices[0] = 0.7 ether;
22     prices[1] = 0.5 ether;
23     prices[2] = 0.3 ether;
24     prices[3] = 0.1 ether;
25   }
26
27  /**
28   * @dev Select a new manager
29   * @param _manager new manager
30   */
31   function transferPriv(address _manager) public payable {
32     manager = _manager;
33 }
```

Nachdem der Besitzer festgelegt ist, werden in der Funktion `Drinks` die Preise für die Produkte festgelegt. Um diese nachträglich zu verändern oder zu ergänzen, kann die Funktion `changePrice` aufgerufen werden (siehe Listing 5.2). Die Anforderung für das Ändern der Preise ist, dass der Absender der Nachricht (`msg.sender`) auch der Manager des Vertrags ist.

Die `calcPrice`-Funktion zur Berechnung des Preises wird unabhängig von der Bezahlfunktion gehandhabt, da sie so auch im voraus vom Nutzer aufgerufen werden kann, um dem Nutzer den zu bezahlenden Betrag anzuzeigen (siehe Listing 5.2). Der Nutzer übergibt die ID und die Anzahl der Produkte als Array an die Funktion. Dies ist für wenige Produkte zwar specheraufwendig, da gegebenenfalls bei einer Bestellung von nur einem Produkt bereits ein Array erzeugt werden muss. Es lässt jedoch zu, dass der Besitzer weitere Produkte zur Getränkekarte hinzufügen kann, ohne den Smart-Contract in seiner Funktionalität zu beeinflussen. In der Funktion selbst werden alle Preise mit der entsprechenden Anzahl an Items verrechnet und die Summe als `unsigend Integer` zurückgegeben. Die

5.2. Smart-Contracts

Funktion wird mit einem sogenannten `view`-Modifier versehen. Das bedeutet, dass die Funktion garantiert, keine Veränderungen an der Blockchain durchzuführen. So entstehen dem Nutzer beim Ausführen der Funktion keine Kosten. Die Ausführung der Funktion muss daher auch nicht explizit durch den Nutzer genehmigt werden.

Listing 5.2: Drinks: price-functions

```

1  /**
2  * @dev Set a new Price
3  * @param _drinks id of the drink
4  * @param _amounts new price of the drink
5  */
6  function changePrice(uint _drink, uint _price)
7  public view returns (uint) {
8      require(msg.sender == manager);
9      prices[_drink] = _price;
10 }
11
12
13 /**
14 * @dev Calculates the correct price of Drinks ordered
15 * @notice Please call the calcPrice-Function first, and send the
16 *         right Amount of Ether
17 * @param _drinks provide array of listed drinks
18 * @param _amounts array of ordered amounts
19 * @return uint, price of cocktails in Ether
20 */
21 function calcPrice(uint[] _drinks, uint[] _amounts)
22 public view returns (uint) {
23     uint price;
24     for (uint i = 0; i < _drinks.length; i++) {
25         price += _amounts[i]* prices[_drinks[i]];
26     }
27     return price;
28 }
```

Der Kern des Vertrages bietet die Bezahlfunktion `takeOrder` für die Produkte (siehe Listing 5.3). Als Parameter der Funktion muss angegeben werden, für welchen Tisch die Bestellung aufgegeben wurde. Die Herausforderung liegt an dieser Stelle, wie bereits bei der Berechnungsfunktion, in der Darstellung und in der Speicherung der bestellten Getränke. Je nach Anzahl der insgesamt verfügbaren Produkte auf der Karte empfiehlt es sich, entweder eine Aufreihung der Anzahl aller Getränke anzugeben, dabei jede ID als einfachen `Integer`-Wert. Jedoch würden dann auch ein `Integer` übergeben werden, falls kein Produkt bestellt wurde. Mit wachsender Anzahl an Produkten auf der Getränkekarte lohnt sich daher ein Mapping. Dies lässt auch ein einfaches Erweitern der Speisekarte zu, kostet aber bei der Ausführung eine minimal höhere Summe an Gas. Da es sich jedoch im Anwendungsfall lediglich um einstellige Centbeträge handeln wird, wird dies als vernachlässigbar betrachtet. Um die Möglichkeit zu gewährleisten, dass auch etwas

5.2. Smart-Contracts

für den Nachbartisch bestellt werden kann, wird die Tisch-ID, die im User-Interface ausgewählt wird, als Primärschlüssel für die Speicherung der Getränke verwendet.

Die Bezahlfunktion nutzt abschließend die Berechnungsfunktion, um zu überprüfen, ob der bezahlte Betrag in Wei dem zuvor berechneten Betrag entspricht. Sollte das nicht der Fall sein, wird die Funktion abgebrochen und das verbleibende Geld wird dem Nutzer zurückerstattet. Danach wird durch alle Getränke durchitteriert und die entsprechende Menge auf der Blockchain hinterlegt.

Auch dem Personal muss die Möglichkeit gegeben werden, die Anzahl der bestellten Getränke abzurufen. Dafür steht die `getOrder`-Funktion bereit (siehe Listing 5.3). Dabei handelt es sich erneut um eine reine `view`-Funktion, da zu ihrer Ausführung die Blockchain nicht verändert werden muss. Mit ihr kann jeweils nur ein Getränk abgerufen werden, da die Rückgabe von dynamischen Arrays und Mappings nicht unterstützt wird [?]. Da die Ausführung der Funktion jedoch kostenlos ist, kann diese Funktion iterativ für jedes Produkt aufgerufen werden.

Ist der Kunde bedient, kann die Bestellung über die `serve`-Funktion wieder von der Blockchain gelöscht werden (siehe Listing 5.3). Auch hier werden wieder zwei Arrays und die ID des Tisches übergeben. Die Anzahl wird hier nicht zurück gesetzt sondern lediglich um angegebene Anzahl verringert, sodass die Bestellung in mehreren Teilen erfolgen kann.

Zuletzt steht die Fallback-Funktion, die aufgerufen wird, wenn versehentlich Geld an den Smart-Contract geschickt wird, ohne eine Funktion aufzurufen. Das Geld kann dann zurückgesendet werden.

Listing 5.3: Drinks: order, server and fallback-functions

```

1  /**
2   * @dev Token-transfer from msg.sender to address
3   * @param _drinks provide array of listed drinks
4   * @param _amounts array of ordered amounts
5   */
6  function takeOrder(uint _table, uint[] _drinks, uint[] _amounts)
7  public payable {
8      uint price = calcPrice(_drinks, _amounts);
9      require (msg.value >= price);
10     for (uint i = 0; i < _drinks.length; i++){
11         orders[_table][_drinks[i]] += _amounts[i];
12     }
13 }
14
15 /**
16  * @dev Returns amount of cocktails ordered from a specific address
17  * @param _table id of table
18  * @param _drinkId id of requested drink
19  * @return uint, amount of drinks
20 */
21 function getOrder(uint _table, uint _drinkId)
22 public view returns (uint){
23     return orders[_table][_drinkId];
24 }
```

5.2. Smart-Contracts

```

25 /**
26 * @dev Removes ordered cocktails from list
27 * @notice Please do not call this contract if you arent the
28     restaurant manager
29 * @param _drinks provide array of served drinks
30 * @param _amounts array of served amounts
31 */
32 function serve(uint _table, uint[] _drinks, uint[] _amounts)
33 public {
34     require(msg.sender == manager);
35     for (uint i = 0; i < _drinks.length; i++){
36         orders[_table][_drinks[i]] -= _amounts[i];
37     }
38 }
39 /**
40 * @notice Fallback. Do not call this function ever. Thanks.
41 */
42 function()
43 public payable {
44     EmptyPay(msg.sender, msg.value);
45 }
```

5.2.1. Entscheidungen für das User Interface

Dem Design des Frontendes wird hierbei eine besondere Rolle zugeteilt. Er muss die Einfachheit des Smart-Contracts durch eine simple aber funktionierende Nutzerschnittstelle ergänzen. Die Blockchain-Technik wird noch lange nicht von jedem Nutzer akzeptiert, deshalb soll die Hemmschwelle besonders niedrig gehalten werden. [30] Um dies zu ermöglichen wurde die Entscheidung getroffen, zwei Front-End-Anbindungen zu erstellen. Der relevante Bestellungs-Anteil des Benutzer-Frontends, welcher die Übersicht der einzelnen Drink-Elemente zur Verfügung stellt, ist dazu in Abbildung 5.1 unten dargestellt.

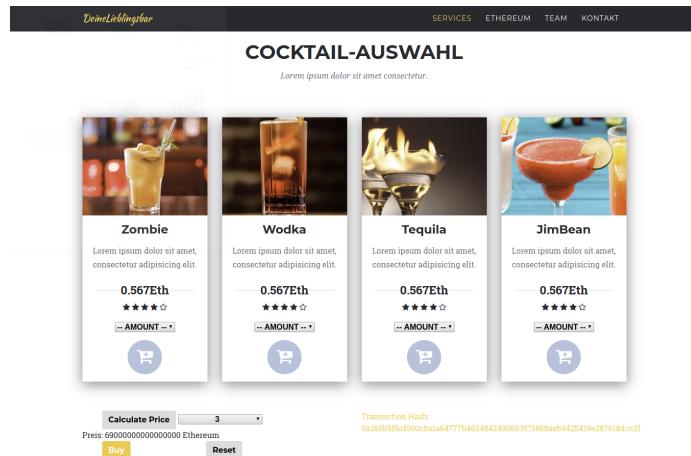
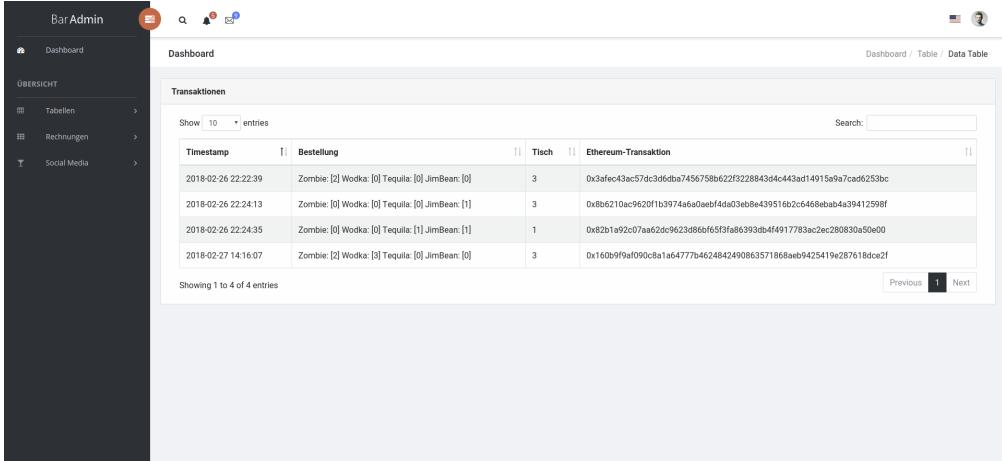


Abbildung 5.1.: Ansicht für Besucher

5.2. Smart-Contracts

Die Frontend-Anbindung des Nutzers ist dabei so gestaltet, dass sich der Nutzer die gewünschten Produkte aus der Menükarte wählen kann und zu einem Warenkorb hinzufügen kann. Hat der Nutzer den Auswahlvorgang abgeschlossen, so kann er über den Menü-Button „Preis berechnen“ die Blockchain-Funktionalität nutzen, um den exakten Preis zu berechnen. Dies wird über die Funktionalität der Blockchain gelöst, da so sichergestellt werden kann, dass exakt dieselben Parameter verwendet werden und die Transaktion nicht fehlschlägt. Ist der Nutzer mit dem Preis einverstanden, kann er die Transaktion eröffnen. Dabei öffnet sich die Metamask Integration, welche den erwarteten Preis bereits anzeigt. Der Nutzer braucht lediglich die Transaktion bestätigen. Um dem Nutzer eine Bestätigung zu geben, wird der Transaktion-Hash angezeigt und auf Etherscan verlinkt [31]. Dort kann er nun überprüfen, ob die Transaktion schon angenommen wurde oder ob diese noch ausstehend ist. Ein Mitarbeiter könnte sich nun die Transaktion einscannen und die Bezahlung überprüfen.

Die Mitarbeiter, welche die Bestellung aufnehmen, und verarbeiten erhalten eine gesonderte Ansicht. Im Hintergrund werden die relevanten Daten der Bestellung, inklusive des Transaction-Hashs, in eine Datenbank eingetragen. Dies darf nur nach erfolgreicher Bezahlung erfolgen. Der Absender der Nachricht ist zwar in der Transaktion hinterlegt, jedoch ist diese Information für die Bestellung nicht weiter relevant, da der entsprechende Tisch angegeben wurde. Als Primary-Key dient der Datenbank eine fortlaufende Nummerierung der Transaktion-Hash. Dieser wird zusätzlich als Link hinterlegt und kann bei Bedarf überprüft werden. Falls ein Fehler unterlaufen sollte, könnte das Personal den tatsächlichen Absender des Geldes ermitteln, vorausgesetzt der Käufer gibt seine Ethereum-Adresse preis. Auch eine manuelle Rückzahlung wäre dadurch möglich. Des Weiteren muss die Auflistung selbstverständlich die bestellten Getränke enthalten. Die Struktur dieser Übersicht ist ausschnittsweise in Abbildung 5.2 unten visualisiert. Die hier dargestellte Administrations-Oberfläche stellt dabei eine Übersicht der im Tagesverlauf generierten Bestellungen bereit, welche die Zuordnung zu den Ethereum-Transaktionen ermöglicht.



The screenshot shows a dark-themed administrative interface titled "Bar Admin". On the left, a sidebar lists "Dashboard", "ÜBERSICHT" (with "Tabellen", "Rechnungen", and "Social Media" options), and "Dashboard". The main area is titled "Dashboard" and contains a table titled "Transaktionen". The table has columns: "Timestamp", "Bestellung", "Tisch", and "Ethereum-Transaktion". It displays four entries from February 26, 2018, and one from February 27, 2018. Each entry includes a timestamp, a list of items ordered (e.g., Zombie: [2] Wodka: [0] Tequila: [0] JimBean: [0]), the table number (e.g., 3 or 1), and the corresponding Ethereum transaction hash. At the bottom of the table, it says "Showing 1 to 4 of 4 entries".

Timestamp	Bestellung	Tisch	Ethereum-Transaktion
2018-02-26 22:22:39	Zombie: [2] Wodka: [0] Tequila: [0] JimBean: [0]	3	0x3afec43ac57dc3dfdb7456738b62f3228843d4c443ad14915e9a7cad6253bc
2018-02-26 22:24:13	Zombie: [0] Wodka: [0] Tequila: [0] JimBean: [1]	3	0x86210ac9620f1b3974a6a0ebf4d9a93eb88439516b2c6459e6ab4a3941299bf
2018-02-26 22:24:35	Zombie: [0] Wodka: [0] Tequila: [1] JimBean: [1]	1	0x82b1e92c07aa62dc9623d86bf65f3fa86393db4f4917783a2ec280830a50e00
2018-02-27 14:16:07	Zombie: [2] Wodka: [3] Tequila: [0] JimBean: [0]	3	0x160b919af090c8a1a64777b4624842490863571868ae9425419e28761bdc02f

Abbildung 5.2.: Ansicht für Mitarbeiter:Bestellungs-Übersicht

5.2. Smart-Contracts

Um aus Kunden- und Betreiber-Perspektive die Interaktion mit der Etherscan-Umgebung zu vereinfachen, können alle Link-Referenzen auf diese Seite direkt in der Bestell- und Administrations-Oberfläche dargestellt werden. Ermöglicht wird diese Funktionalität durch die Verwendung einer internen *Modal*-Komponente, welche zur Echtzeit-Darstellung von Website-Inhalten genutzt werden kann. Durch Angabe des Transaktions-Hashes wird dabei die korrespondierende Etherscan-Übersicht der Transaktion geöffnet. Auf diese Weise können einzelne Transaktions-Parameter unmittelbar innerhalb der Bestell- und Administrationsumgebung verifiziert werden, sodass alle an der Transaktion beteiligten Instanzen jederzeit deren Status abrufen können. Dies ermöglicht insbesondere Anwendern mit limitierten Ethereum-Erfahrungen einen vereinfachten Einstieg in die Verwendung der implementierten Demonstrator-Anwendung. Eine exemplarische Darstellung von Transaktionen innerhalb der *Modal*-Komponente ist in Abbildung 5.3 unten visualisiert.

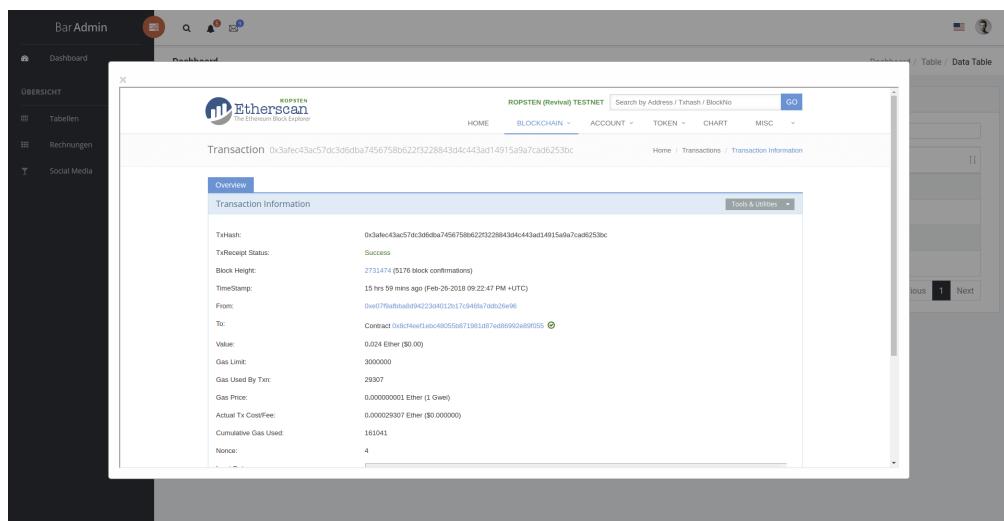


Abbildung 5.3.: Ansicht für Mitarbeiter: Verifikation durch Etherscan

6. Evaluation

Um die Bewertungen des Szenarios und die daraus abgeleiteten Prioritätenschätzungen aus Technologie und Anwendbarkeit nachvollziehen zu können, wurden mit verschiedenen Gastronomien Experteninterviews durchgeführt, bei denen diese über das Thema Blockchain und dem Einsatz in der Gastronomie befragt und darüber hinaus qualitative Fragen zu den einzelnen Einflussfaktoren entsprechend der Anwendbarkeit und dem Nutzen beantworteten. Die Experteninterviews dienen somit auch zur Plausibilitätsprüfung des generierten Modells und der subjektiven Bewertungen der Befragten. Damit kann überprüft werden, ob das Modell gültig, in der vorliegenden Form annehmbar und nachvollziehbar ist [26].

6.1. Durchführung der Interviews

Die Frageformulierung und der Aufbau des Fragebogens erfolgen in Form eines strukturierten Leitfrageninterviews. Dabei wurden die wichtigsten Themen als Fragen formuliert, aber bewusst Raum für tiefer gehende Fragen gelassen. Die nachfolgende Tabelle 6.1 zeigt die vordefinierten Themen, die das Interview behandelt und auf welche näher eingegangen werden kann.

Tabelle 6.1.: Themen für die Experteninterviews

Blockchain	Grundlagen Blockchain kryptografischer Hintergrund Wissenstand Kryptowährung
derzeitige Situation	Durchführung Bestellvorgang Abwicklung Bezahlung Abrechnung nach Bezahlung Trinkgeld Webauftritt
Szenario	Vorstellung des Szenario Änderung am System Vor-/Nachteile Anwendbarkeit für Betrieb

6.1. Durchführung der Interviews

In besonderem Maße sind die Antworten zum Thema der Anwendbarkeit wichtig für die Auswertung, weshalb dort auf eine detaillierte Begründung Wert gelegt wird. Folgende Betriebe im Bereich Gastronomie wurden befragt:

- Unicasino Unibw, Neubiberg
- Taverne Artemis, Ottobrunn
- Die 2, Neubiberg
- La Ciacolada, Neubiberg
- Santorini, Naumburg
- Landgasthof alter Bahnhof, Rohrberg

Alle Betriebe befinden sich in Dörfern oder Kleinstädten. Während *Die 2* hauptsächlich als Bar dient und das *Unicasino* eine separate Bar besitzt, sind die restlichen 4 Gastronomien ausschließlich Gaststätten beziehungsweise Restaurants. Über Kundenzahl und Umsatz sind keine Informationen bekannt.

Nachdem alle Daten erhoben wurde, wurde eine Reduktion auf die wichtigsten Aussagen getroffen, welche weiter untersucht werden. Beim Thema Blockchain fiel sofort auf, dass es eine starke Diversifikation im Wissenstand der Befragten gab. Während die Betreiber von *Die 2* ein etwas umfangreicheres Wissen über Blockchain und Kryptowährungen besaßen, einer von ihnen beschäftigt sich auch mit der Kryptowährung Ethereum, interessiert die Befragten des *Unicasinos* und der Taverne *Artemis* das Thema ein wenig und sie verstehen die Blockchain im groben. die restlichen 3 Betriebe haben nur durch Nachrichten die Begriffe schon einmal gehört. Alle sechs Betriebe assoziieren den Begriff der Blockchain fast ausschließlich nur mit Kryptowährungen und spekulativen Investments. Eine große Gemeinsamkeit aller Betriebe ist die derzeitige Situation in Bezug auf Bestellung, Bezahlung und der Abwicklung von Rechnungen. Die Bestellung läuft durch einen Kellner, der persönlich an den Tisch geht und die Kunden nach ihren Wünschen fragt. Dabei wird eine Speisekarte in Form eines Buches oder einzelnen Blattes dem Kunden überreicht, indem die möglichen Speisen und Getränke inklusive ihrer Preise stehen. Niemand nutzt digitale Produkte für ihre Bestellung oder die Anzeige von Sonderangeboten. Die Bezahlung erfolgt mittels Bargeld oder auf Wunsch per EC-Karte. Laut den Aussagen der Betreiber bevorzugen jüngere Menschen häufiger den Bezahlvorgang mittels EC-Karte. Diese Aussage ist aber nicht nachgewiesen, sondern nur das Empfinden der Befragten. Beim Thema Webauftritt wissen alle Befragten um die Bedeutung und Wichtigkeit. Allerdings besitzen nur 4 Betriebe eine eigene Website. Das *Santorini* und der *Landgasthof alter Bahnhof* haben nur eine kurze Beschreibung auf Drittanbietern. Bei der Beschreibung des Szenarios konnten sich alle Betriebe in die Situation hineinversetzen. Allerdings konnten nur das *Unicasino* und *Die 2* technisch folgend und das Konzept wirklich verstehen. Für die restlichen vier Betriebe war es ein System für die ferne Zukunft. Ein weiterer Aspekt für die meisten, der gegen einen Einsatz sprach, war die geringe Nutzung der Blockchain für echte Transaktionen. Da in

6.2. Risikoanalyse

ihrem Empfinden der Hauptaugenmerk auf die Spekulation von Währungen lag, würden wenig Kunden das System zum Bezahlen nutzen wollen. *Die 2 und die Taverne Artemis* finden das Konzept für Werbezwecke sinnvoll, wenn es kostengünstig ist. Das *Unicasino* hält es nicht für umsetzbar, da es weder einen produktiven Beitrag leistet und noch einen effektiven Nutzen bringt. Die restlichen drei Betriebe halten die Umsetzung zwar für möglich, aber den Einsatz nicht für sinnvoll, nicht mal zu Werbezwecken.

Abschließend kann man sagen, dass das Thema einer breiten Masse der Bevölkerung bekannt ist. Allerdings kennen sich gerade in ländlichen Gebieten zu wenig wirklich mit der Technologie und dem Konzept der Blockchain aus. Die Befragten wurden bewusst aus Bevölkerungsarmen Gegenden gezogen um ein gewisses Maß an Vergleichbarkeit zu ermöglichen. Auf Grundlage dieser Ergebnisse können weitere Untersuchungen hinsichtlich einer quantitative Analyse zum Wissenstand über Blockchain durchgeführt werden.

6.2. Risikoanalyse

In diesem Abschnitt erfolgt eine kurze Risikoanalyse des Projektes. Es werden wichtige und realistische Risiken beleuchtet und kurz bewertet. Zunächst werden Risiken betrachtet, welche innerhalb des Betriebs auftreten können. Das erste Risiko stellt dabei die Bereitschaft der Kunden dar. Durch den veränderten Bestell- und Bezahlvorgang müssen sich alle Kunden anpassen. In der Anfangsphase stellt dies für den Kunden sowohl eine erhöhte technische Herausforderung für den Vorgang dar, als auch ein erhöhter Zeitaufwand. Da die Bestellung nun über eine Weboberfläche abläuft, muss diese zuerst mit einem mobilen Gerät aufgerufen werden und danach die gewünschten Produkte ausgewählt und bestätigt werden. Dies stellt für viele, besonders ältere Menschen, eine Hürde, da sie mit dem herkömmlichen Bestellvorgang über einen Kellner vertraut sind und es für den Kunden weniger aufwendig ist. Um dieses Risiko zu minimieren, ist ein möglichst einfach gehaltenes Design zu verwenden. Auch könnte der Betreiber in der Anfangsphase für die Kunden das Produkt gemeinsam bedienen, damit sich die Kunden mit dem System vertraut machen können. Eine Möglichkeit, den Kellner bei Fragen zur Bedienung des Systems zu rufen, sollte eingerichtet sein. Ein nächster möglicher Schwachpunkt ist der Verlust des Kontaktes zum Kunden. Da die Bestellung und die Bezahlung nun online durchgeführt wird, sieht der Kunde den Kellner nur noch beim Bringen der Produkte. Dies kann zur Verminderung der Identifikation zum Geschäft führen. Außerdem können die Kellner keine Vorschläge mehr geben oder auf besondere Wünsche der Kunden eingehen. Dieses Risiko kann dadurch gemindert werden, dass der Kellner zusätzlich um die einzelnen Tische herum geht und den Kontakt zum Kunden sucht. Dies vermindert allerdings den Vorteil, dass sich der Kellner voll auf seine Tätigkeiten wie das Mixen und Bringen von Getränken konzentrieren kann.

Als nächstes wird die technische Sichtweise betrachtet und die dortigen Risiken aufgezeigt. Das größte Risiko ist dort die Abhängigkeit vom Internet und Server. Ohne eine funktionierende Internetverbindung ist der Webserver nicht erreichbar, wodurch keine Bestellungen abgewickelt werden können. Des Weiteren muss der Server voll funktions-

6.2. Risikoanalyse

fähig sein und auch vor Angriffen und Schwachstellen geschützt werden. Dieses Risiko kann zu einem Stopp des Bestellungsvorgang führen, was dem Betrieb schadet. Es kann allerdings einfach behoben werden, indem der Betrieb auch weiterhin ohne Abhängigkeit von der Blockchain durchführbar ist und somit eine Backup-Lösung vorhanden ist.

Die nächsten Punkte befassen sich mit externen Einflüssen auf das Projekt. Ein bedeutender Einfluss ist die Bekanntheit des Themas Blockchain. Durch die ständigen Nachrichten in letzter Zeit ist das Thema sehr in den Fokus der Medien gerückt [24] und damit auch für viele Kunden ein möglicher Grund, das Geschäft allein aufgrund des Produktes besuchen zu wollen. Sollte der Trend um das Thema abschwächen, so werden sich auch weniger Menschen für die Blockchain und ihre Produkte interessieren. Damit ist das Produkt sehr abhängig von der gesamten Entwicklung der Blockchain. Das letzte hier betrachtete Risiko ist der Einfluss von Staaten und anderen Organisationen. Regulierungen auf die Blockchain werden seit dem Jahr 2018 mehr und mehr diskutiert [25]. Diese Regularien können zur Folge haben, dass eine große Bürokratische Hürde für kleinere Geschäfte gelegt wird, die den Nutzen des Produktes abschwächt und damit unrentabel macht. Allerdings können Regulierungen auch zu einer breiten Akzeptanz führen, da einheitliche Prozesse und Richtlinien für den Umgang mit der Blockchain eingeführt werden. Staatliches Eingreifen in das Thema ist abschließend schwer abzuschätzen und damit sollte es für die Risikobewertung vorläufig vernachlässigt werden. Es bedarf weiterer Untersuchungen und erste Entscheidungen von staatlichen Organisationen um den Einfluss korrekt beurteilen zu können.

Abschließend zeigt die Grafik 6.1 eine SWOT-Analyse des Szenarios. Die SWOT-Analyse (Akronym für Strengths, Weaknesses, Opportunities und Threats) ist ein Instrument der strategischen Planung und dient der strukturierten Zusammenfassung der wichtigsten Aspekte eines Produktes hinsichtlich dessen Position im wirtschaftlichen Kontext.

6.2. Risikoanalyse

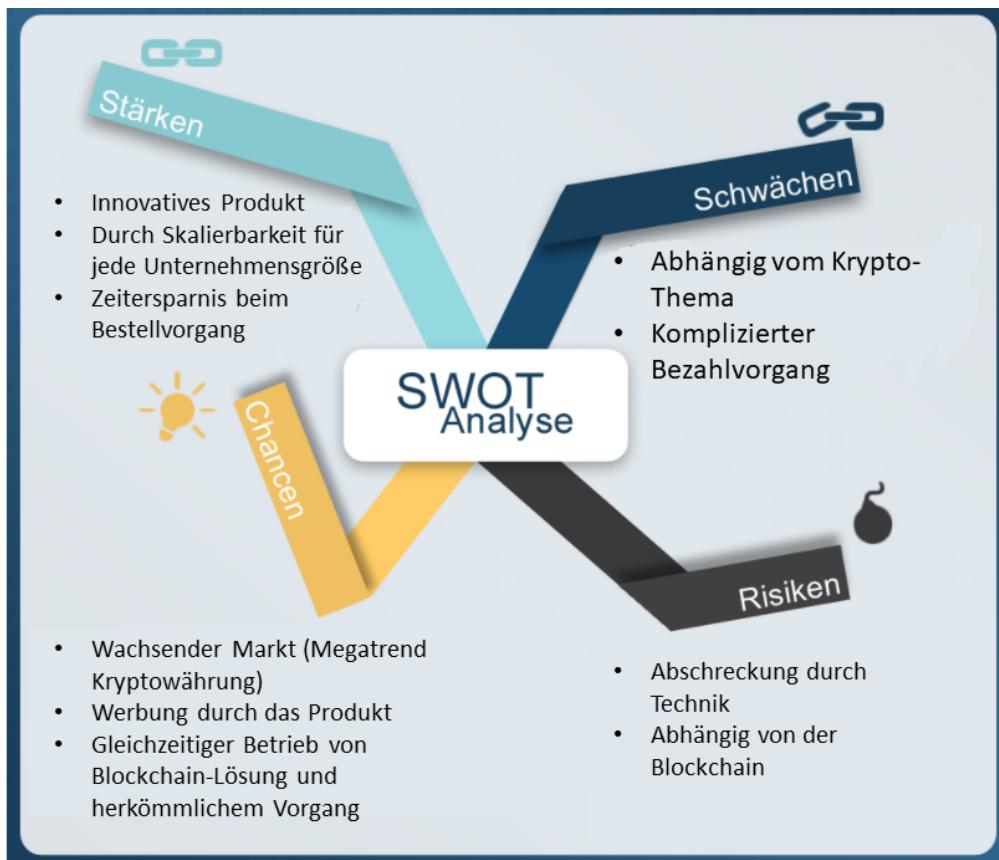


Abbildung 6.1.: SWOT-Analyse der Protolösung

7. Konklusion

Im Folgenden wird zunächst der Inhalt der Arbeit zusammengefasst 7.1. Hierbei werden gegebenenfalls auch entsprechende Ergebnisse aufgezeigt. Anschließend werden in Abschnitt 7.2 potentielle Ausblicke in zukünftige Verwendung gegeben beziehungsweise Vermutungen geäußert, welche Anwendungen die Technologie zukünftig ermöglichen könnte.

7.1. Zusammenfassung

In Kapitel 1.1 wurde zunächst an das Thema herangeführt. Hierzu zählte die schemenhafte Einordnung des Wertes von Daten sowie das anschließende Begründen anhand eines Beispiels, warum solche Daten wichtig sind und es die Möglichkeit geben sollte diese auf Korrektheit zu prüfen. Außerdem wurde die Grundfunktionalität einer Blockchain kurz angerissen und die wohl bekanntesten Vertreter genannt. In Kapitel 2 wurden die Grundlagen erläutert. Hierzu gehörten der Begriff des Krypto Hashes, der Unterschied zwischen zentralisierten und verteilten beziehungsweise Peer-to-Peer Netzen, das beleuchten von public-/privat-Key Verfahren beziehungsweise asymmetrische Kryptographie sowie die Einordnung der wohl bekanntesten Kryptowährungen. Dabei handelte es sich um Bitcoin und Ethereum, welche als Abschluss des Kapitels verglichen wurden. In Kapitel 3 wurde die methodische Herangehensweise der Arbeit vorgestellt. Dies beinhaltete sowohl gruppeninterne organisatorische Aspekte, als auch zur Durchführung der Arbeit notwendige beziehungsweise geplante Schritte. Zusätzlich wurden die Grundlagen beziehungsweise Hintergründe der einzelnen Aspekte kurz beleuchtet. Gruppenintern betraf dies beispielsweise das Kanban-Tool Trello, welches die Gruppe verwendete, um die Arbeiten zu planen und Einsicht in den Stand anderer Teilarbeiten zu haben sowie permanent einschätzen zu können in welchem Arbeitsstadium sich die Gruppe grade befand. Eher auf das Projekt selbst bezogen waren eine Machbarkeitsstudie ob die Idee tatsächlich so umsetzbar sei und eine Risikoanalyse. Die tatsächliche Umsetzung dieser Maßnahmen folgten im nächsten Kapitel. Bei diesem Kapitel handelt es sich um Kapitel 4. Hier wurde zuerst ein betriebswirtschaftlicher Kontext hergestellt beziehungsweise erläutert. Darauf folgte die Darstellung des Fallbeispiels Whoppercoin Burger King. Der Hauptteil des Kapitels besteht der Beschreibung von Use Cases im Gastronomie-Kontext. Dort wurden neben der Erläuterung der Kunden- und Betreiberperspektive auch die erwähnte Machbarkeitsstudie und Risikoanalyse durchgeführt. Das Kapitel endet schließlich mit einem Interview mit der UniBar. Im folgenden und letzten Inhaltenkapitel 5 wurde die Entwicklung eines Prototypen auf Webbasis beschrieben. Hierfür wurden außerdem die Grundlagen beziehungsweise die Bedeutung von Smart Contracts erläutert. Im Verlauf des Kapitels wird kurzer Einblick in den Quellcode gegeben sowie grob die Funktionalität

7.2. Ausblick

der Applikation beschrieben. Das Kapitel schließt mit bildlichen Darstellungen der App beziehungsweise des Frontends.

7.2. Ausblick

In dieser Arbeit ist es gelungen einen vorläufigen Prototypen zu entwerfen, der geforderte Funktionalitäten simulierend abbilden kann. Auch die Machbarkeitsstudie viel positiv aus. Somit wurde sowohl theoretisch als auch ansatzweise praktisch gezeigt, dass eine tatsächliche Umsetzung möglich sein sollte. Es ist durchaus denkbar, dass bei der eventuellen Weiterführung in zukünftigen Projekten eine verwertbare beziehungsweise einsetzbare Software produziert werden könnte. Grade für kleinere Unternehmen biete es sich an solche dezentralen Lösungen zu wählen, da sie das Unternehmen selbst entlasten würden, was aber nicht heißt, dass die Technologie für große Unternehmen keinen Mehrwert bringen könnte. Bei der vorgestellten Technologie handelt es sich um brandaktuelle Technologie beziehungsweise in einigen Forschungseinrichtung behandelte Konzepte. Dies bedeutet allerdings, dass man sich, um das Thema zumindest ansatzweise erfassen zu können ausgiebig damit auseinandersetzen muss. Hierfür haben viele Menschen aber einfach nicht die Zeit und befinden sich zusätzlich in einem komplett anderen Tätigkeitsfeld. Im Rahmen der Arbeit wurde Interviews durchgeführt. Bei diesen wurde die eben geäußerte Vermutung bestätigt, dass den Betroffenen das nötige Fachwissen fehlt um den potentiellen technischen Mehrwert erfassen zu können. Wenn die Technologie sich etablieren soll ist es sinnvoll darüber nachzudenken vorher erst einmal eine Art technische Aufklärung zu betreiben, damit die Betroffenen wissen, womit sie es wirklich zu tun haben und was es ihnen bringen könnte diese Technologie zu verwenden. In dieser Arbeit wurde exemplarisch die Gastronomie ausgewählt, womit es sich um einen alltäglichen Lebensbereich vieler handelt. Die aufgezeigten Funktionalitäten lassen sich allerdings auf viele andere Bereiche, in welchen Transaktionen durchgeführt werden, übertragen und die Technologie ist natürlich nicht auf diese Sparte beschränkt. Um ein Beispiel zu nennen, welches jetzt nicht, wie das in dieser Arbeit gelieferte Beispiel, direkter Teil der Gastronomie wäre, könnte beispielsweise ein Logistikunternehmen, was hier den Zulieferer abbilden könnte herangezogen werden. Wenn die Geschäfte zwischen Zulieferer und Restaurant oder Ähnlichem nun mit einer ähnlichen Technologie, wie im Demonstrator gezeigt wurde, Verträge abschließen würde wäre es beispielsweise nicht mehr notwendig, dass tatsächlich unterschriebene Verträge existieren. Dies würde gegebenenfalls den Postweg, oder die tatsächliche Anwesenheit der beiden Vertragsparteien nicht weiter benötigen, da diese Verträge auch aus der Entfernung abgeschlossen werden könnten. Im spezifischen Kontext der Gastronomie könnte beispielsweise der Bestellvorgang nahezu komplett automatisiert werden. Die Kunden bestellen per App, welche diese an die Betreiber übermittelt. Dieser kann nun den Wünschen des Kunden entsprechen und die Bestellung zustellen lassen. Der Vorteil wäre hierdurch das einsparen von Bedienungspersonal, was finanzielle Vorteile hat. Außerdem wird die Fehlerquelle von falschen Notizen oder Vergesslichkeit der Bedienung beseitigt. Die Anwendungsfälle sind lediglich durch die Kreativität eingeschränkt. Ob es dazu kommt kann man weder bestätigen, noch widerlegen. Ein potentieller Nutzen

7.2. Ausblick

der Technologie wurde aufgezeigt. Es ist zumindest denkbar, dass das ein oder andere Unternehmen sich dessen in zukünftigen Projekten bedienen wird.

A. Transkript der Interviews

7.2. Ausblick

Ablauf Interview:

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsgebiete?)
2. Erklärung der Blockchain, Erklärung von Ethereum
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
4. Vorstellung des Szenarios
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
6. Abschließende Worte

7.2. Ausblick

Landgasthof alter Bahnhof (Rohrberg)

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsbereiche?)
 - Name schon mal gehört
 - Kein Wissen über genaue Bedeutung und Anwendung
2. Erklärung der Blockchain, Erklärung von Ethereum
 - Kurze Erklärung über die Technologie und Anwendungsmöglichkeiten
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
 - Herkömmlicher Bestellvorgang
 - Speisekarte als Buch, indem Speisen und Getränke mitsamt Preisen stehen
 - Bar und Kartenzahlung möglich
 - Webauftritt über diverse Drittanbieter, keine eigene Homepage
4. Vorstellung des Szenarios
 - Kurze Erklärung des Szenarios
 - Erläuterung der Vor- und Nachteile
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
 - Interessantes Thema
 - Könnte eines Tages die Zahlungsmethode ändern
 - Heute nicht vorstellbar (zu wenig wissen darüber, kaum jemand nutzt es)
 - Für Werbezwecke aufgrund der genannten Gründe nicht nutzbar
 - Kosten-Nutzen nicht positiv
6. Abschließende Worte
 - ländliche Gegend folgt selten neuen Trends
 - setzt auf altbewährtes

7.2. Ausblick

Taverne Artemis (Ottobrunn)

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsbereiche?)
 - Im groben informiert
 - Durch Nachrichten in den Fokus geraten
 - Für zu kompliziert erachtet
2. Erklärung der Blockchain, Erklärung von Ethereum
 - Kurze Erklärung über die Technologie und Anwendungsmöglichkeiten
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
 - Herkömmlicher Bestellvorgang (Kellner geht zum Tisch und nimmt persönlich Bestellung auf)
 - Speisekarte als Buch, Extrablatt für Sonderangebote/Tageskarte
 - Bar und Kartenzahlung möglich, junge Menschen häufiger mit EC-Karte
 - Eigener Webauftritt mit kleiner Homepage
4. Vorstellung des Szenarios
 - Kurze Erklärung des Szenarios
 - Erläuterung der Vor- und Nachteile
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
 - Interessantes Thema, aber sehr komplex und nur für technikaffine
 - Heute nicht vorstellbar (zu wenig wissen darüber, kaum jemand nutzt es in der Gegend)
 - Für Werbezwecke vielleicht nutzbar, wenn keine Kosten entstehen
6. Abschließende Worte
 - Gerne mit fertigem Produkt wiederkommen
 - Wenn Kryptothema interessant bleibt, wird es viel verändern

7.2. Ausblick

Die 2 (Neubiberg)

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsgebiete?)
 - Kryptowährungen meist eine eigene Blockchain, funktioniert dann ein virtueller "Ledger", also eine Art Bestandsbuch, in dem jede Transaktion gespeichert wird
 - hervorragend als dezentrale Währung
 - direkte Verbindung zwischen allen Teilnehmern des Netzwerkes aufgebaut und somit ist es einer einzelnen Institution nicht möglich, den Geldfluss zu korrumpern
 - Anwendungsgebiete besonders beim "Digital Cash und Micro-Transaktionen"
 - Durch Blockchain kann Geld relativ schnell, zügig und sicher Benutzer wechseln
2. Erklärung der Blockchain, Erklärung von Ethereum
 - entfällt, da umfangreiches Wissen vorhanden
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
 - normaler Bestellvorgang
 - möglich mit EC-Karte und Bargeld zu bezahlen
 - mehr Kunden nutzen die Kartenzahlung, da Bargeld öfter vergessen
 - umständlicher als mit Bargeld, da der Bezahlvorgang mit dem EC-Kartengerät wesentlich länger dauert
 - eigener Webauftritt
4. Vorstellung des Szenarios
 - Kurze Erklärung des Szenarios
 - Erläuterung der Vor- und Nachteile
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
 - viele Menschen vertrauen dem Blockchain-Prinzip nicht (Kursschwankungen)
 - Marketing Technisch könnte es natürlich vorteilhaft sein, wenn man zu den ersten "gehört"
 - Art Alleinstellungsmerkmal - "Cool"
 - Wenn Kosten für Anschaffung nicht zu hoch sind, hört sich das auf jeden Fall interessant an
6. Abschließende Worte

7.2. Ausblick

- definitiv nicht, dass die Blockchaintechnologie in naher Zukunft das Bargeld ablöst.
- Bargeld auch verwenden wenn der Akku mal leer ist oder man kein Netz hat

7.2. Ausblick

Unicasino (Neubiberg)

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsgebiete?)
 - durch Whitepaper Bitcoin/Blockchain und Ethereum und News einigermaßen 'tiefgehendes' Verständnis der Technologie/technischen Komponenten
 - Blockchain: Kohärente, Konsistenz- und Integritätsichernde Transaktionstechnologie
 - Kryptowährungen: Basierend auf Blockchain dezentrale, partiell anonymisierbare Transaktionen möglich;
 - Verzicht auf zentrale Vertrauensinstanz (wie beispielsweise einer Bank)
 - soll Regulierung und Kontrolle erschweren
2. Erklärung der Blockchain, Erklärung von Ethereum
 - Kurze Erklärung über die Technologie und Anwendungsmöglichkeiten
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
 - Bezahlung über Bargeld oder EC-Karte
 - Unterschiedliche Bestellungs-/Bezahlvorgänge (Restaurant und Bar)
 - Bar erst bezahlen, dann Getränk
 - Restaurant erst Essen/Trinken, am Ende bezahlen (Trinkgeldaspekt)
 - Bestellvorgang manuell, Bargeld eventuell hinderlich/unhandlich
 - Website für allgemeine Informationen, Werbung für Veranstaltungen und Onlinereservierungen
4. Vorstellung des Szenarios
 - Kurze Erklärung des Szenarios
 - Erläuterung der Vor- und Nachteile
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
 - Da Mobilgerät immer dabei, Bezahloption naheliegend
 - Mit Wallet auf Handy Transaktionen einfach zu realisieren
 - Gerade im aktuellen Blockchain-Hype Potential nutzbar ('Hipster'-Bar)
 - Webauftritt heute eigentlich obligatorisch, Integration Bezahlfunktionalität sinnvoll
 - Feinheiten der Umsetzung schwierig (Wie findet erste Authentifizierung des Nutzers statt?)

7.2. Ausblick

- Umsetzbarkeit wohl nur aus marketing-technischer Sicht sinnvoll, Konzept bringt weder Betreiber noch Endkunde signifikante Komfort-Vorteil
- Umsetzung nur als Trend-Konzept, Technologie liefert keinen produktiver Beitrag

6. Abschließende Worte

- Blockchain-Technologie in der praktischen Verwendbarkeit stark limitiert
- Insbesondere Skalierbarkeit und Stabilität der Währungs-Anteile kritisch

La Ciacolada (Neubiberg)

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsbereiche?)
 - Kryptowährungen sind Online Währungen, die unfälschbar sind
 - Nur Bitcoin aus den Medien
 - Blockchain ist die Technologie, die hinter Bitcoin steht
 -
2. Erklärung der Blockchain, Erklärung von Ethereum
 - Kurze Erklärung über die Technologie und Anwendungsmöglichkeiten (aufbauend auf dem vorhandenen Wissen)
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
 - Website für allgemeine Informationen, Werbung für Veranstaltungen und Onlinereservierungen
 - Kellner nimmt die Bestellung auf und notiert diese auf einem Zettel (also alles analog)
 - kennt im Regelfall die Preise und rechnet den Gesamtpreis dann ab
 - Vertrauensbasis gegenüber den Kellnern gearbeitet, da diese leicht betrügen könnten
 - Bezahlung wird regulär entweder mit Bargeld oder Bankkarte
4. Vorstellung des Szenarios
 - Kurze Erklärung des Szenarios
 - Erläuterung der Vor- und Nachteile
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
 - Heute nicht vorstellbar (zu wenig wissen darüber)
 - jetziges System funktioniert
 - theoretisch aber umsetzbar, da jeder Smartphone besitzt
6. Abschließende Worte
 - keine

7.2. Ausblick

Santorino (Naumburg)

1. Fragen zur Blockchain (Wissensstand: Was sind Kryptowährungen? Was ist die Blockchain? Anwendungsbereiche?)
 - bereits schon etwas über Kryptowährungen gehört
 - Kryptowährungen in den letzten Jahren großer Hype
 - aber auch Beginn der Regulierung bzw. Verbot oder strikterer Umgang durch Banken/Regierungen
 - Begriff Blockchain nur in Zusammenhang mit Kryptowährungen gehört
2. Erklärung der Blockchain, Erklärung von Ethereum
 - Kurze Erklärung über die Technologie und Anwendungsmöglichkeiten (aufbauend auf vorhandenem Wissen)
3. Fragen über die derzeitige Situation der Bar (Bestellvorgang, Bezahlung, Webauftritt)
 - Bedienung kommt zum Kunden, nimmt Bestellung auf
 - Getränke/Speisen auf Zettel beim Kunden festgehalten
 - Bezahlung vor Verlassen des Lokals in bar oder mit EC-Karte
 - kein eigener Webauftritt vorhanden, auch nicht notwendig (Stammkunden)
4. Vorstellung des Szenarios
 - Kurze Erklärung des Szenarios
 - Erläuterung der Vor- und Nachteile
5. Fragen über das Szenario: Vorstellbar, Heute Umsetzbar (Begründung für ja und nein)
 - nicht vorstellbar, dass es zum Einsatz kommt
 - Lokal ist auf dem Dorf, viele ältere Gäste
 - die wenigsten kennen sich damit aus
 - Technik müsste angeschafft werden
 - kein Mehrwert dadurch
6. Abschließende Worte
 - keine

Literaturverzeichnis

- [1] ZEIT ONLINE GmbH. Das Gold des 21. Jahrhunderts. <http://www.zeit.de/angebote/zukunfts faktor-technologie/gold-21-jahrhundert>, Stand:Februar. 2018.
- [2] Statista GmbH. Die Top 10 der Kryptowährungen. <https://de.statista.com/infografik/1939/marktkapitalisierung-von-kryptowaehrungen/>, Stand:Februar. 2018.
- [3] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE*, 2016.
- [4] Bundesanstalt für Finanzdienstleistungsaufsicht. Blockchain-technologie, 2017.
- [5] Elektronik Kompendium. Kryptografische hash-funktionen.
- [6] Elektronik Kompendium. Sha und md.
- [7] Wiki Commens. Network topology star.
- [8] Wiki Commens. Network topology mesh.
- [9] William Stallings. *Network Security Essentials: Applications and Standards*. Prentice Hall Press, 2010.
- [10] Anton Badev and Matthew Chen. Bitcoin: Technical background and data analysis, 2014.
- [11] Pedro Franco. *Understanding Bitcoin: Cryptography, Engineering, and Economics*. John Wiley and Sons, Ltd, 2014.
- [12] Annalee McWilliams. Bitcoin, ethereum und co.: Kryptowährungen einfach erklärt, 2017.
- [13] Vincenzo Morabito. *Business Innovation Through Blockchain*. Springer Verlag, 2017.
- [14] Bitcoin-Wiki. Bitcoin-wiki.
- [15] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.
- [16] blockchain.info. Bitcoin blockchain size.
- [17] Bitcoin-Wiki. Proof-of-work.
- [18] Chjango Unchained. Consensus compare: Tendermint bft vs. eos dpos.
- [19] Vitalik Buterin. A next-generation smart contract and decentralized application platform.

Literaturverzeichnis

- [20] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger.
- [21] Stephan Tual. Ethereum launches.
- [22] Ethereumkaufen. Ethereum zukunft: Der ethereum fahrplan.
- [23] Dirk Stelzer. Risikoanalysen als hilfsmittel zur entwicklung von sicherheitskonzepten in der informationsverarbeitung. *ITSicherheitsmanagement in Banken.*, 2002.
- [24] Wiwo:. <http://blog.wiwo.de/look-at-it/2017/11/16/8-von-10-unternehmen-kennen-blockchain-aber-nur-7-prozent-halten-technologie-fuer-marktreif/>, zuletzt besucht am 04.03.2018.
- [25] Markus Kaulartz. <https://causa.tagesspiegel.de/wirtschaft/wie-weiter-mit-den-bitcoins/ein-bisschen-regulierung-reicht-schon.html>, zuletzt besucht am: 04.03.218.
- [26] Jochen Gläser. *Experteninterviews und qualitative Inhaltsanalyse*. Springer, 2010.
- [27] Ethereum. *Solidity Documentation*, 2017.
- [28] Gavin Wood. *Ethereum: A secure decentralised generalised transaction ledger*, 2017.
- [29] Ethereum White Paper. *Ethereum White Paper*, 2017.
- [30] Jakob Nielsen. *Designing Web Usability: The Practice of Simplicity*. New Riders Publishing, Thousand Oaks, CA, USA, 1999.
- [31] Etherscan. The ethereum block explorer, 2018.
- [32] Chris Dannen. *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*. Apress, Berkely, CA, USA, 1st edition, 2017.
- [33] Wiwo:. <https://www.wiwo.de/adv/capgemini/bitcoin-technologie-unternehmen gehen-auf-tuchfuehlung-mit-blockchain/19815578.html>, zuletzt besucht am 21.02.2018.
- [34] Faizod:. <https://faizod.com/studie-zum-thema-blockchain/>, zuletzt besucht am: 20.02.2018.
- [35] Vincent Schlatt, AndrÃ© Schweizer, Nils Urbach, and Gilbert Fridgen. Blockchain: Grundlagen, anwendungen und potenziale. *Projektgruppe Wirtschaftsinformatik des Fraunhofer-Instituts für Angewandte Informationstechnik FIT*, 2016.