Trusted Blockchains für das offene, intelligente Energienetz der Zukunft

tbiEnergy

Abschlussbericht

Zuwendungsempfänger:	Hochschule Bremen
Verbundvorhaben:	tbiEnergy
Teilvorhaben:	IT-Sicherheit für Trusted Blockchains, im intelligenten Energienetz der Zukunft (ITSitbiE)
Förderkennzeichen:	03EI6029B
Laufzeit des Vorhabens:	01.06.2020 – 31.05.2023
Berichtsdatum:	27.11.2023
Firma:	Hochschule Bremen
Anschrift:	Flughafenallee 10 28199 Bremen
Telefon:	+49 421 5905 5483
Ansprechpartner:	Prof. Dr. Richard Sethmann
E-Mail:	Richard.Sethmann@hs-bremen.de

Gefördert durch:

Bundesministerium
für Wirtschaft
und Klimaschutz

aufgrund eines Beschlusses
des Deutschen Bundestages

Autoren

HOCHSCHULE BREMEN

Prof. Dr. rer. nat. Richard Sethmann

Giacomo Gritzan, M. Sc.

Michelle Jakobi, B. Sc.

Sibille Knodel, M. Sc.

Torben Petrow, B. Sc.

Inhalt

Abbildungsverzeichnis

Tabellenverzeichnis

I. Schlussbericht – Kurzdarstellung

In diesem ersten Teil des Schlussberichtes wird ein kurzer Überblick gegeben, auf welchen Grundideen das tbiEnergy-Projekt basiert und mit welchen Annahmen und Lösungsansätzen an die Umsetzung des Forschungsvorhabens herangegangen wurde. Dabei wird der Fokus auf die Vorgehensweise der Hochschule Bremen gelegt und weiterhin der aktuelle wissenschaftlich-technische Stand der Forschung beschrieben.

1 Überblick

Das Projekt tbiEnergy setzt auf dem bestehenden regulierten Energiemarkt auf und adressiert mit einem ganzheitlichen Blockchain-Ansatz die momentan in den Energienetzen anstehende Problematik der effektiven Integration alternativer Energieerzeugung in bestehende Netzstrukturen. Außerdem soll die aktuelle Lücke der fehlenden Komfortdienste des heutigen intelligenten Stromnetzes durch den Einsatz der Blockchain geschlossen werden. Mit Hilfe der Blockchain-Technologie und den in ihr formulierten Smart-Contracts, lassen sich innovative Geschäftsmodelle auch ohne hohe Investitionen in die IKT oder Softwareinfrastruktur bei gleichzeitiger inhärenter Sicherheit realisieren. Am Markt befindliche Blockchain-Lösungen sind allerdings bisher nicht explizit für den energiewirtschaftlichen Einsatz konzipiert. Es fehlen Kundenschnittstellen, eine Anbindung an die Infrastruktur der Energieversorger sowie ein stringentes Hardwaresicherheitskonzept, vergleichbar mit dem durch das Bundesamt für Sicherheit in der Informationstechnik (BSI) gesetzten Standard für den Betrieb digitaler Kommunikationstechnologie kritischer Infrastrukturen. Über das Mehrwertdienstekonzept des deutschen Smart-Meter-Gateways lassen sich die Vorstellungen des BSI und die Vorzüge einer kryptographisch gesicherten, verteilten Datenbank in tbiEnergy zusammenführen. Neben dem Kundennutzen findet sich ein weiterer Vorteil in der Eichrechtskonformität und regulatorischen Integrierbarkeit bereits im Energienetz vorhandener Flexibilitäten oder regenerativer Energiequellen von Energieversorgern. Dazu finden sich im Konsortium Hardwaresicherheitsexperten wie die Infineon AG, die devolo AG als Hersteller von Smart-Grid-Hardware und -Lösungen, die Hochschule Bremen als ausgewiesene Expertin im Thema IT-Sicherheit, dem Blockchain-Startup Arxum GmbH und die Stadtwerkte Trier AöR als Anwendungspartner zusammen. Durch die Mischung der breit aufgestellten fachlichen Kompetenzen soll eine Plattform geschaffen werden, die generische Geschäftsprozesse innerhalb einer Blockchain abbildbar macht, und sich dennoch in die aktuellen energiewirtschaftlichen Regularien einfügt. Ein Novum ist im Besonderen der lückenlose Einsatz von Hardwaresicherheitsmechanismen unter Anwendung ressourcensparender Konsensmechanismen. Ein weiterer Forschungsgegenstand des Projektes ist der Einsatz teilprivater Blockchains (konsortiale Blockchains), sowie eine möglichst reibungslose Integration von sicheren Benutzerschnittstellen.

Die im Rahmen des Projektes erarbeiteten Konzepte kulminieren in einem abschließenden Demonstrator, der im Rahmen eines Feldtests anhand eines Testkatalogs überprüft wird.

2 Aufgabenstellung

Im Folgenden wird die Ausgangslage und das daraus resultierende Problem geschildert, welches die Grundlage für das Forschungsprojekt tbiEnergy bietet. Zusätzlich werden noch die Ziele der Hochschule Bremen (HSB) im Zuge des Forschungsprojektes aufgeschlüsselt.

2.1 Ausgangslage und Problemschilderung

Die Gesamtvorhabenbeschreibung (GVB) stellt dar, dass im Zuge des Klimawandels und der Energiewende die Energienetze vor großen Herausforderungen stehen, die es zu lösen gilt. Die Energiebranche strebt daher die Einführung sogenannter intelligenter Energienetze (engl. Smart Grids) an, wodurch aktuelle energienetzspezifische Probleme gelöst werden und sich den Endverbrauchern

neue Möglichkeiten eröffnen. Ein Ansatz ist z. B. die lokale Vermarktung von Energie mithilfe der Anbindung intelligenter Messsysteme wie Smart Meter Gateways (SMGWs) an eine Blockchain. Dadurch wird eine Vermarktung des Stroms per Smart Contracts ermöglicht. So eröffnen sich neue Möglichkeiten wie z. B. der Verkauf selbst erzeugter Energie in die direkte Nachbarschaft. Auch die Koordination zwischen verschiedenen Akteuren wie Energiespeichern, Flexibilitäten (z. B. Wärmepumpen oder Heißwasserspeichern) und weiteren Marktgegenständen ist möglich. Durch all diese Anwendungsfälle entsteht allerdings auch eine komplexere Kommunikationsinfrastruktur als sie klassischerweise im Energienetz zu finden ist – und bringt damit ein deutlich vergrößertes Bedrohungspotential mit sich. Diesen Bedrohungen gilt es frühzeitig unter der Verwendung entsprechender IT-Sicherheitstechnologien entgegenzutreten. [1]

Wie in der GVB erläutert, wurde die Idee eines lokalen blockchainbasierten Marktes bereits in mehreren anderen wissenschaftlichen Projekten untersucht und in der Praxis erprobt. tbiEnergy setzt auf der bestehenden Forschung auf und adressiert dabei im Gegensatz zu bisherigen Ansätzen stärker die Aspekte der IT-Sicherheit, der Energieeffizienz, der eingesetzten Konsensus-Algorithmen, der Absicherung des Identitätsmanagements sowie die damit einhergehenden Regulierungskonformitäten.

Gerade der Einsatz von Hardwaresicherheitsmodulen (HSM) wird in bestehenden Forschungsansätzen nicht ausreichend betrachtet. Das Projekt legt den Fokus daher auf die Hardwaresicherheit standardisierbarer Lösungen für lokale Marktplätze und grenzt sich damit deutlich von den bisherigen wissenschaftlichen Projekten ab. Es erfolgt dabei eine Anlehnung an sogenannte „Hardwarewallets“ („Hardwaregeldbörsen“) wie sie bereits für Kryptowährungen eingesetzt werden. Diese bereits aus der Sicherheitstechnik bekannten HSMs bieten die Möglichkeit, kryptographische Schlüssel zu verwalten und sich um die Verschlüsselung zu kümmern. In einem solchen Szenario ist es vorstellbar, dass die Entitäten eines Peer-to-Peer-Netzes (P2P) direkt über eine Blockchain-Infrastruktur miteinander kommunizieren und nur im Bedarfsfall bei rechenintensiven Operationen die Unterstützung des SMGWs benötigen. Außerdem lässt sich durch die Verwendung hardwarebasierter Schlüssel die Vertrauensstufe zwischen den teilnehmenden Entitäten drastisch steigern und somit gerade die Sicherheit von Konsens-Algorithmen, die auf die Identifizierung ihrer Teilnehmer setzen, stärken. Dies ermöglicht es, die Vertrauenslücke im Übertragungsweg zwischen der als sicher geltenden Blockchain und der Erzeugung der zu übertragenden Daten zu schließen. Die Einbindung von HSMs in Smart-Contracts und die Prüfung der Integration der HSMs in den Konsensus-Prozess können hierbei als weitere Forschungsgegenstände gesehen werden. Durch die Verwendung der Blockchain-Technologie in direkter Kombination mit hardwaregesicherten „Internet of Things“-Geräten (IoT) / SMGWs wird eine unkontrollierte Einflussnahme durch Drittparteien unterbunden.

2.2 Zielsetzung der Hochschule Bremen im Forschungsprojekt

Die HSB setzt ihren Fokus auf den Bereich der IT-Sicherheit der zu entwickelnden Lösung. Im Zuge der Bedrohungsanalyse sollen vorhandene Gefahren erkannt werden und in das nach dem „Secure by Design“-Prinzip entwickelte Konzept einfließen. Das Projektziel für die HSB ist es somit, ein speziell auf Energienetze zugeschnittenes Blockchain-Konzept zu entwickeln, zu implementieren und die anschließende Inbetriebnahme zu unterstützen. Dabei sollen die Konsensfindung und die Identitätsverwaltung durch den Einsatz von HSMs effizient abgesichert und unterstützt werden. Im Rahmen des Verbundprojekts tbiEnergy soll somit ein Demonstrator für eine effiziente Blockchain- und hardwarebasierte IT-Sicherheitslösung entwickelt werden, der in intelligenten Energienetzen eingesetzt werden kann und die speziellen Charakteristika eines zukünftigen Energiemarktes berücksichtigt (siehe Abbildung 1).

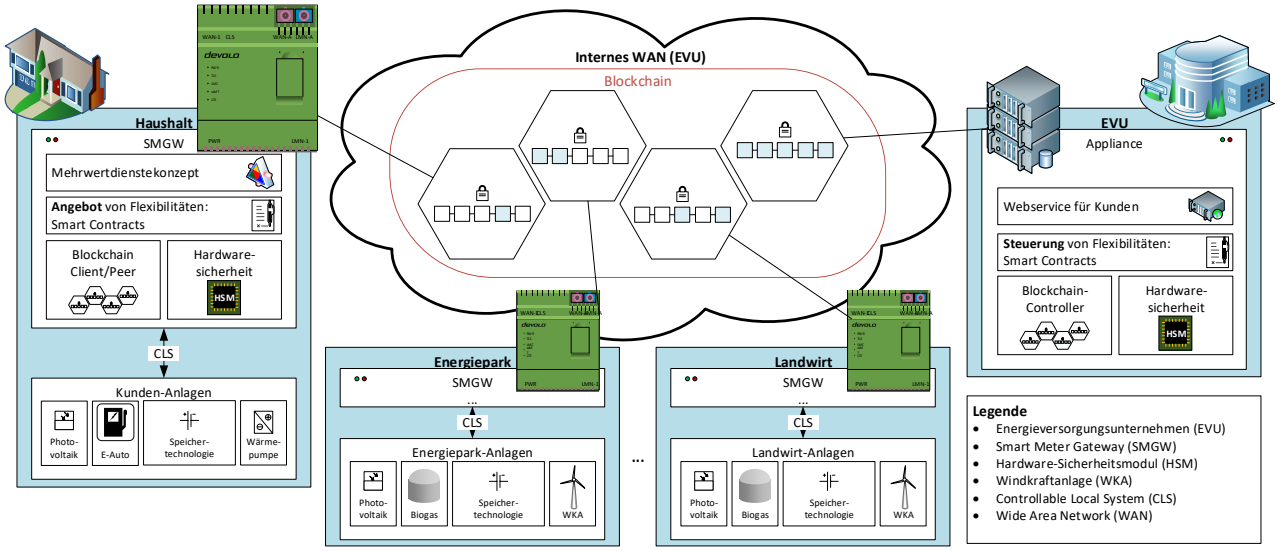

Abbildung 1: Schematische Darstellung des Lösungsansatzes

3 Voraussetzungen, unter denen das Vorhaben durchgeführt wurde

Das Forschungsvorhaben tbiEnergy wurde im Rahmen des 7. Energieforschungsprogramm durch das Bundesministerium für Wirtschaft und Klimaschutz (BMWK, ehemals BMWi) gefördert. Das Forschungsvorhaben adressierte dabei den Förderaufruf „Digitalisierung der Energiewende“. Die HSB betrachtete hierbei den Schwerpunkt „Digitalisierung der Energiewende“ mit Fokus auf „Modellprojekte“. Außerdem wurden die Querschnittsthemen „Erschließung neuer Märkte“, „Ladeinfrastruktur“ und „Innovative und verbesserte Technologien und Schutzkonzepte“ berücksichtigt.

4 Planung und Ablauf des Vorhabens

Das Forschungsvorhaben wurde in enger Zusammenarbeit mit den Konsortialpartnern erarbeitet. Im Folgenden wird auf die geplanten Tätigkeiten der HSB und den tatsächlich erfolgten Ablauf des Vorhabens eingegangen. Abbildung 2 gibt einen Überblick über die Struktur der Arbeitspakete welche die Aufgaben innerhalb des Projekts strukturieren und nachfolgend jeweils kurz erläutert werden.

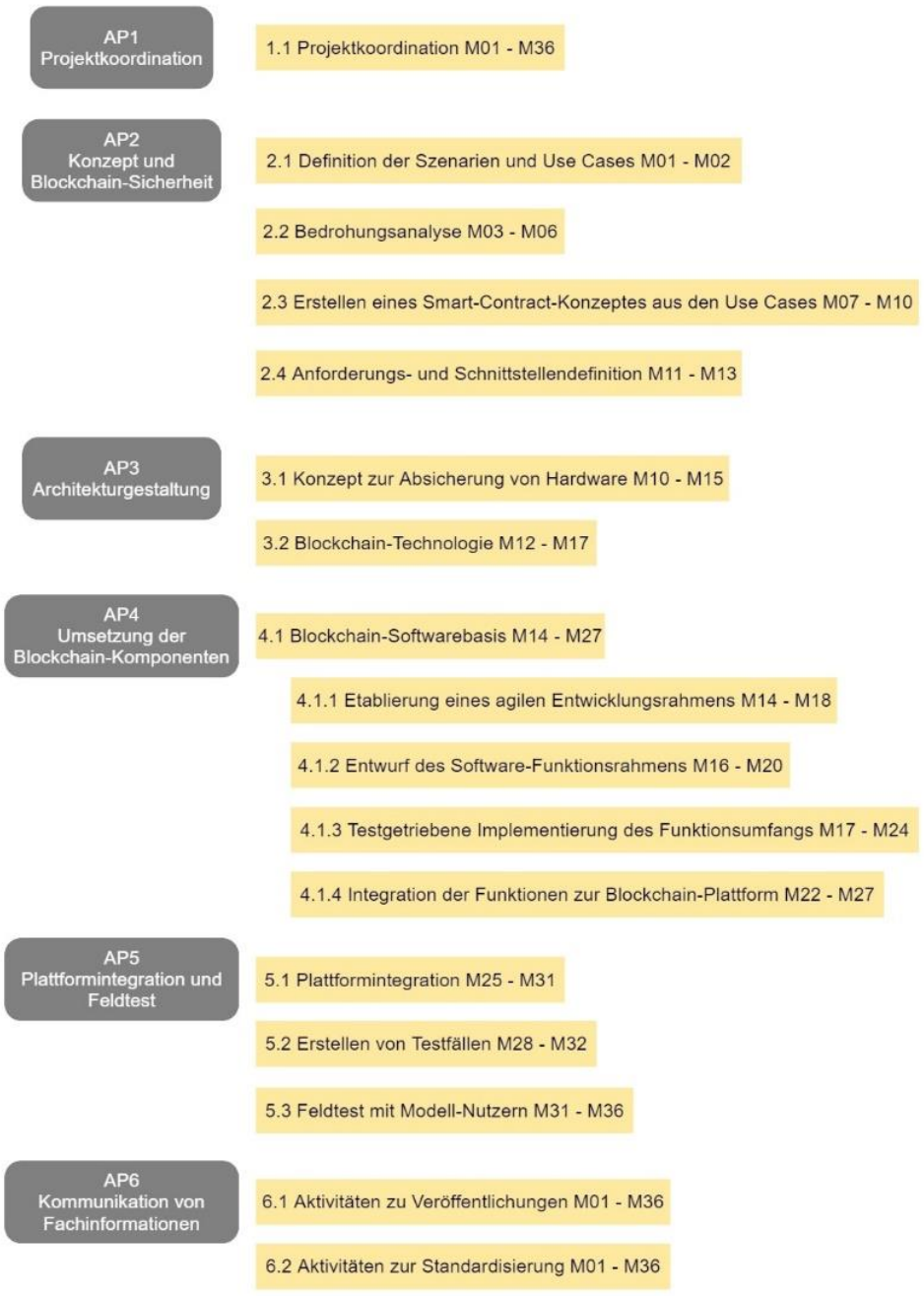

Abbildung 2: Übersicht Arbeitspakete

Im Rahmen der Projektkoordination (AP 1) wurden die Aufwände zur Koordination der Zusammenarbeit zwischen den Partnern zusammengefasst. Entsprechend der erfolgten Planung hat die HSB im Rahmen des Arbeitspakets z. B. an der Organisation von Workshops und Projekttreffen mitgewirkt und die Konsortialpartner unterstützt.

Die Hochschule Bremen leitete das Arbeitspaket „Konzept und Blockchain-Sicherheit“ (AP 2). Zu Beginn des Arbeitspakets wurden gemeinsam in enger Abstimmung mit den Projektpartnern Anwendungsszenarien sowie die abzubildenden Use Cases erarbeitet. Im Sinne des „Secure by Design“-Prinzips wurde noch vor der eigentlichen Konzeptionsphase eine Bedrohungsanalyse für die zu verwendenden Blockchain-Technologien und die Smart-Grid-Hardware durchgeführt, um die gewonnenen Erkenntnisse beim Design und der Entwicklung zu berücksichtigen. Bei der Bedrohungsanalyse wurden bestehende Empfehlungen wie die des BSI berücksichtigt. Unter Berücksichtigung der identifizierten Use Cases und der sich aus der Sicherheitsanalyse ergebenen Randbedingungen wurden die Anwendungsfälle in abstrakt formulierte „Smart Contracts“ übersetzt. Damit einhergehend erfolgte die Auswahl des Konsensus-Algorithmus „Delegated Proof of Stake“ (DPoS). Abgeleitet aus den Szenarien und der Bedrohungsanalyse wurde eine ganzheitliche Architektur entwickelt, die alle logischen und physischen Schnittstellen beinhaltet. Dabei wurde die Authentizität und Sicherheit aller Komponenten entlang der Wertschöpfungskette berücksichtigt.

Basierend auf der in AP 2 durchgeführten Bedrohungsanalyse wurden in AP 3 Blockchain-Technologien miteinander verglichen und eine für den Einsatz in Energienetzen geeignete Blockchain-Technologie ausgewählt. Als Ergebnis des Arbeitspakets konnte mit EOS[1] eine Blockchain-Technologie ausgewählt werden, die es ermöglicht, zur Steigerung der Sicherheit das HSM zu integrieren und die im Projekt priorisierten Schutzziele zu erfüllen. Außerdem wurden die in AP 2.3 identifizierten Konsensus-Algorithmen auf ihre Anwendbarkeit mit hardwarebasierten Sicherheitstechnologien evaluiert.

AP 4 umfasste unter anderem die notwendigen Implementierungen für die gewählte Blockchain-Technologie. Außerdem erfolgte die Integration der Hardwaresicherheit und die Entwicklung der für den Demonstrator notwendigen Microservicearchitektur und des dazugehörigen Webfrontends. Für die SMGW-Hardware wurde ein entsprechender Software-Stack zu Ankopplung an die ausgewählte Blockchain erarbeitet. Im Rahmen des Teilarbeitspakets AP 4.1.1 hat die HSB in Zusammenarbeit mit den Projektpartnern, die an der Softwareentwicklung beteiligt waren, festgelegt, welche Technologien, Programmiersprachen, Code-Konventionen und Toolchains zum Einsatz kommen sollten. Weiterhin wurde die Einhaltung dieser Rahmenrichtlinien mithilfe von Continuous Integration sichergestellt. Als Ergebnis konnte eine entsprechende gemeinsame Entwicklungsumgebung mit den beteiligten Partnern realisiert werden.

In Teilarbeitspakets AP 4.1.2 begann die HSB mit der testgetriebenen Implementierung einzelner Module und legte zunächst den Funktionsumfang basierend auf der Architektur fest. Dies umfasste die Ableitung von Funktionsclustern aus der Architektur sowie die Aufschlüsselung dieser Funktionscluster in Funktionen und Module. Zudem wurden die Kommunikationsschichten definiert. In enger Zusammenarbeit mit den Projektpartnern leitete die Hochschule Bremen gemeinsam mit den Projektpartnern fortlaufend Funktionscluster unter Berücksichtigung gängiger IT-Sicherheitsaspekte und den Voraussetzungen der im Rahmen von AP 3.2 ausgewählten Blockchain-Technologie ab. Auf dieser Basis wurden feingranulare Softwareentwürfe für die in den Funktionsclustern enthaltenen Module und Funktionen in iterativer Weise erstellt.

Im Rahmen von AP 4.1.3 wurden die zuvor in AP 4.1.2 definierten Softwareentwürfe durch die HSB in iterativer Weise implementiert. Diese Umsetzung der Komponenten erfolgte testgetrieben und wurde durch automatisierte Komponententests und Integrationstests fortlaufend innerhalb einer CI/CD Umgebung überprüft. Mit Hilfe dieser automatisierten Tests konnte eine möglichst fehlerfreie Implementierung durch die Hochschule Bremen gewährleistet werden.

[1] URL: https://eos.io/

Im AP 4.1.4, dem letzten Teilarbeitspaket des AP 4, unterstützte die HSB die Projektpartner fortlaufend bei dem Entwurf der zu implementierenden Smart Contracts und der Integration der Systemkomponenten und deren Funktionen zur Blockchain-Plattform. Von der HSB wurde die Schnittstelle zu den umgesetzten Smart Contracts in die entwickelte Microservice-Architektur integriert und damit die Funktionen der Systemkomponenten ins Gesamtsystem integriert.

Das AP 5 vereinte die zuvor erarbeiteten Systemkomponenten mit einer Plattformintegration im AP 5.1 vollständig mit dem Gesamtsystem. Zudem wurden die Arbeiten anschließend mit einem Feldtest in AP 5.3 überprüft, welcher die Beantwortung des zuvor im AP 5.2 erstellten Testkatalogs mit sich brachte. Im Zuge der Plattformintegration wurden die Systemkomponenten der HSB in die AWS-Infrastruktur des Projektpartners Arxum eingebunden. Mithilfe von CI/CD-Pipelines konnten neue Funktionen und Anpassungen mit geringem Aufwand umgesetzt und veröffentlicht werden. Diese Anpassungen wurden in dem Teilarbeitspaket durch die HSB nach den Praktiken der agilen Softwareentwicklung evaluiert und die notwendige Anpassung umgesetzt.

Im Zuge des AP 5.2 wurden mit den Projektpartnern Testfälle basierend auf die in AP 2.1 enthaltenen Sollwerte erstellt und in einem Testkatalog zusammengefasst. Im AP 5.3 wurden dann die Ist-Werte des entwickelten Demonstrators im Feldtest mit Modell-Nutzern mit den Soll-Werten verglichen. Der Feldtest verlief positiv und alle Testfälle konnten erfüllt werden. Vorbereitend dazu hat die HSB den Projektpartnern entsprechende Deployment-Skripte zur Unterstützung bei der Ausbringung der Softwarekomponenten auf die Hardware der Feldtestteilnehmer zur Verfügung gestellt. Das Webportal wurde ebenfalls für eine erleichterte Fehlerbehebung angepasst, um eine Übersicht der angeschlossenen Hard- und Softwarekomponenten und deren Erreichbarkeit zu erhalten.

Die HSB trug ebenfalls einen Beitrag zum AP 6 betreffend der Kommunikation von Fachinformationen bei. Im Rahmen des AP 6.1 wurde durch die HSB das Abstract auf der 6. Blockchain Autumn School (BAS2022) zum Thema „A blockchain based local energy market" [2] eingereicht und akzeptiert. Anschließend wurde ein entsprechendes Full Paper eingereicht, veröffentlicht und präsentiert. Die Veröffentlichung wurde außerdem mit dem zweiten Platz des Best Paper Award ausgezeichnet. Die HSB wirkte zusätzlich im Rahmen des AP 6.2 an der Standardisierung der Interoperabilität von DLT-Systemen in der ISO/TC 307 „Blockchain and distributed ledger technologies"[2] in der „Working Group 7 – Interoperability" mit.

[2] URL: https://www.iso.org/committee/6266604.html

5 Wissenschaftlich-technischer Stand

In der GVB werden bereits die wesentlichen Punkte aus Wissenschaft und Technik vorgestellt. Darüber hinaus wurden die folgenden weiteren Punkte für das Teilvorhaben der Hochschule Bremen betrachtet.

5.1 Analyse von Bedrohungen in der IT

Für die Bedrohungsanalyse gibt es verschiedene Herangehensweisen, die meist in Kombination verwendet werden, um ein möglichst genaues Bild der Bedrohungslage zu entwickeln. Eine Auswahl verschiedener Möglichkeiten wird im Folgenden erläutert.

5.2 Systematisches Vorgehen

Unter einem systematischen Vorgehen wird die Erstellung einer Bedrohungsanalyse mithilfe bestehender Bedrohungskataloge verstanden. Organisationen wie das Bundesamt für Sicherheit in der Informationstechnik (BSI) geben hierzu z. B. ein *Kompendium* [3] mit Bedrohungen heraus. Meistens werden neben den Bedrohungen auch Gegenmaßnahmen in den Bausteinen des Kompendiums erfasst, sodass diese direkt abgeleitet und später auf ihre Umsetzung hin geprüft werden können. Im Fall der Blockchain gibt es aktuell keinen entsprechenden IT-Grundschutz-Baustein, der sich auf den Anwendungsfall der „Blockchain“ bezieht. Das BSI empfiehlt aber in einer Analyse [4] der Blockchain-Technologie einige Bausteine des IT-Grundschutzes, die betrachtet werden sollten.

5.3 Kreative Verfahren

Das kreative Vorgehen steht dem systematischen Vorgehen gegenüber. Bei den kreativen Verfahren werden strukturgebende Maßnahmen wie der Bedrohungsbaum oder die Bedrohungsmatrix eingesetzt. Dabei werden strukturelle Verläufe eines Angriffs ausgehend von einem Angriffsziel und ggf. einem Angriffsprofil aufgebaut.

5.4 Kategorisierung nach STRIDE

STRIDE [5] ist eine Methode, die durch das Unternehmen Microsoft entwickelt wurde und bei der dortigen Softwareentwicklung eingesetzt wird, aber auch in der Industrie und Wissenschaft ihre Anwendung findet. Mithilfe von STRIDE können verschiedene Bedrohungen im Rahmen einer Bedrohungsanalyse in sechs Klassen kategorisiert werden. Anhand dieser Klassen ist es möglich, über eine korrespondierende Klassifizierung von Gegenmaßnahmen, geeignete Maßnahmen zur Verhinderung der kategorisierten Bedrohungen auszuwählen. Der Name der Methode ist hierbei ein Akronym aus den Oberbegriffen der sechs berücksichtigten Bedrohungsklassen:

- **S – Spoofing Identity**: Diese Klasse bezeichnet Bedrohungen, mit denen Mechanismen zur Authentifizierung oder Identifikation untergraben werden können. Als mögliche Gegenmaßnahmen kommen Methoden zur Authentifizierung infrage.
- **T – Tampering with Data**: Diese Klasse bezeichnet Bedrohungen, die darauf abzielen, persistente sowie transportierte Daten unberechtigt zu ändern oder zu ersetzen. Als mögliche Gegenmaßnahmen kommen Methoden zum Schutz der Integrität infrage.
- **R – Repudiation**: Diese Klasse von Bedrohungen nutzt Systeme aus, bei denen es nur schwer möglich ist, einzelne Aktionen auf einen Nutzer als Verursacher zurückzuführen. Als mögliche Gegenmaßnahmen kommen Methoden zur Überwachung der Nutzeraktivität infrage, um die Nichtabstreitbarkeit bei gewissen Aktionen sicherzustellen.
- **I – Information Disclosure**: Diese Klasse enthält Bedrohungen, bei denen es darum geht, trotz fehlender Berechtigungen auf Informationen zuzugreifen oder sie während des Transports zu

lesen. Als mögliche Gegenmaßnahmen kommen Methoden zum Schutz der Vertraulichkeit infrage.

- **D – Denial of Service**: Diese Klasse beinhaltet Bedrohungen, mit denen erreicht werden soll, dass Berechtigte nicht mehr auf für sie freigegebene Ressourcen zugreifen können. Als mögliche Gegenmaßnahmen kommen Methoden zum Schutz der Verfügbarkeit infrage.
- **E – Elevation of Privilege**: Die Bedrohungen dieser Klasse ermöglichen den Zugriff auf Ressourcen, die normalerweise geschützt sind, indem über Sicherheitslücken höhere Privilegien durch Nutzer mit geringeren Privilegien erreicht werden. Als mögliche Gegenmaßnahmen kommen Methoden zur Autorisierung infrage.

Zur Durchführung einer Bedrohungsanalyse mit STRIDE wird ein System oder eine Software meist in einzelne Komponenten unterteilt. Hierbei sollten zumindest externe Abhängigkeiten und Schnittstellen, die auftretenden Nutzertypen, Zugriffspunkte, zu schützende Werte und Vertrauensstufen für den Zugang über einen Zugriffspunkt und den Zugriff auf einen zu schützenden Wert, festgehalten werden. Mit Hilfe dieser Informationen können Datenflüsse und Zusammenhänge in einer Anwendung oder einem System nachvollzogen werden. Daraufhin können die gesammelten Informationen einzeln und in Kombination auf Bedrohungen analysiert werden. Hierzu werden systematische und kreative Verfahren (s. o.) eingesetzt.

6 Zusammenarbeit mit anderen Stellen

Die Forschungsgruppe Rechnernetze und Informationssicherheit (FRI) der Hochschule Bremen (HSB) ist im Institut für Informatik und Automation tätig. Kernthemen sind u. a. Trusted Computing, sichere Zugangs- und Zugriffsverfahren, Vertrauensstellungen in komplexen Systemen und Fachkenntnisse im Bereich der Netztechnik. Die Hochschule Bremen hat bereits diverse Forschungsprojekte im Bereich Informationssicherheit in Energienetzen gemeinsam mit einem Großteil der Projektpartner durchgeführt und ist zudem aktuell an einem weiteren Forschungsprojekt im Energie-Umfeld beteiligt. Im Forschungsprojekt SPIDER wurde ein SMGW nach BSI-Vorgaben für das intelligente Energienetz entwickelt. In einem weiteren Forschungsprojekt wurde die Systemsicherheit von Energieversorgungsnetzen bei der Einbindung von Informations- und Kommunikationstechnologien (SEnCom) untersucht. Im Forschungsprojekt „Sichere Datenkommunikation für die verteilte Fabrik der Zukunft" (SiDaFab) wurden Trusted-Computing-Konzepte für ein Smart Gateway I4.0 untersucht und weiterentwickelt. Insbesondere der Einsatz eines TPM 2.0 für Measured Boot und Remote Attestation wurde untersucht und das notwendige Know-how an die Forschungspartner der Industrie vermittelt. Im Rahmen des SiDaFab-Projekts hat sich die HSB an der DIN SPEC 27070 beteiligt und ihre Expertise eingebracht. Dort wurde das Ziel verfolgt, eine Referenzarchitektur für ein Security Gateway für den Austausch von Industriedaten und Diensten zu schaffen. Ein anderes Forschungsprojekte der Hochschule Bremen befasste sich mit der Entwicklung von „Methoden für Energienetzakteure zur Prävention, Detektion und Reaktion bei IT-Angriffen und -Ausfällen“ (MEDIT). In einem weiteren Forschungsprojekt werden „Polymorphe Agenten als querschnittliche Softwaretechnologie zur Analyse der Betriebssicherheit von cyber-physischen Systemen“ entwickelt (PYRATE). Dafür treffen in einer Simulationsumgebung KI-gestützte Agenten auf Energienetze, den Energiemarkt, Energie-IKT und versuchen, diese zu stören, um daraus Erkenntnisse für eine sichere Betriebsführung ableiten zu können.

II. Schlussbericht – Eingehende Darstellung

Der zweite Teil dieses Schlussberichtes konzentriert sich auf die praktische Umsetzung der in Teil I beschriebenen Konzepte und Inhalte. Dabei wird u. a. die Implementierung der Softwareanteile genauer betrachtet.

1 Definition der Szenarien und Use Cases

Im Rahmen von AP 2.1 hat die Hochschule Bremen ihr Know-how im Bereich der IT-Sicherheit eingebracht und konnte zusammen mit den Projektpartnern das Arbeitspaket wie im Projektplan vorgesehen abschließen. Die Hochschule hat als AP-Leiter die im Rahmen des APs notwendigen Aufgaben ausgearbeitet, koordiniert und entsprechende Vorlagen zur Erfassung der Inhalte zur Verfügung gestellt. Im ersten Schritt wurde ein gemeinsames Verständnis für die Prozesse auf Basis der Blockchain geschaffen und eine Liste möglicher Use Cases erstellt. Anschließend wurden die erfassten Use Cases gemeinsam priorisiert und die Ausarbeitung auf die zuständigen Projektpartner verteilt. Dabei wurden vier Use Cases erstellt:

- Use-Case_01_Lokaler-Energiemarkt
- Use-Case_02_Bilzanzkreispool
- Use-Case_03_Power2XToken
- Use-Case_P2P-Handel

Die Hochschule Bremen hat dabei hauptverantwortlich den Use Case des lokalen Energiemarkts (siehe Abbildung 3) erarbeitet und die Partner bei der Erarbeitung der anderen priorisierten Use Cases unterstützt. Zusätzlich zu den Haupt-Use-Cases sind zwei überspannende Use Cases erarbeitet worden: Der „Use Case Hardwaresicherheit“ und der „Use Case sichere Kommunikation im Smart-Grid“.

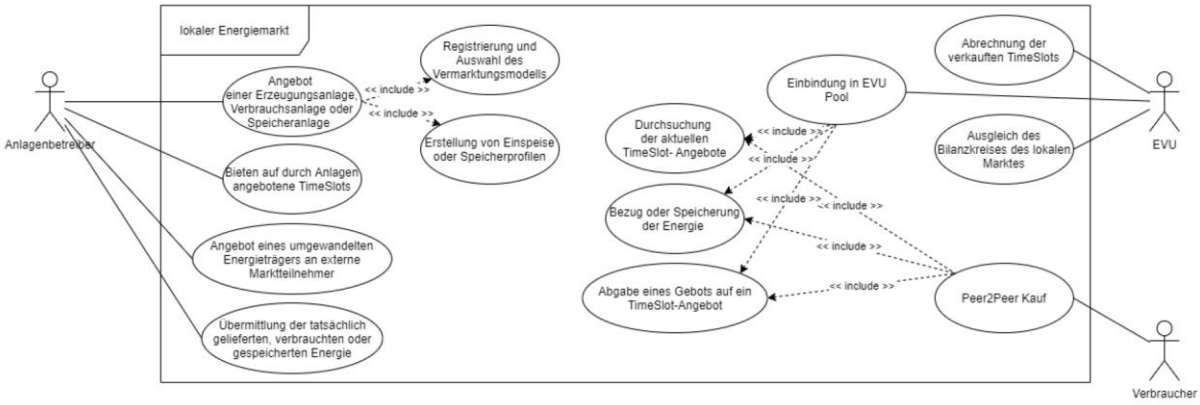

Abbildung 3: Use-Case-Diagramm lokaler Energiemarkt

Um ein Verständnis für die notwendigen Prozesse und Abläufe zu erlangen, wurden die zentralen Tätigkeiten aus den Use Case-Diagrammen in entsprechende Aktivitätsdiagramme überführt. Zusätzlich wurden die fachspezifischen Begrifflichkeiten und Abläufe des lokalen Energiemarkts unter den Partnern mit Hilfe eines gemeinsamen Wording-Dokumentes abgestimmt.

2 Bedrohungsanalyse

Im Rahmen von AP 2.2 wurden im ersten Schritt die im AP geplanten Themenfelder zwischen den Projektpartnern harmonisiert und der formale Rahmen zur späteren Erstellung der Bedrohungsanalyse abgestimmt. Um einen Überblick über die für die Bedrohungsanalyse relevanten Assets zu erlangen,

wurde ein gemeinsames Strukturdiagramm erarbeitet und anschließend je Partner die dazugehörigen Teil-Datenflussdiagramme erstellt. Im Anschluss erfolgte die Zusammenführung der Teildatenflussdiagramme in ein Gesamt-Datenflussdiagramm (siehe Abbildung 4), in dem alle Prozesse, Systeme, Datenflüsse und Entitäten mit individuellen IDs versehen wurden. Die aus AP 2.1 abgeleiteten Ergebnisse dienten dabei als Grundlage zur Identifikation der für das Gesamtsystem notwendigen Assets. Als Methodik für die Erarbeitung der Bedrohungen wurde das von Microsoft entwickelte STRIDE-Modell (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege) ausgewählt. Anschließend wurden den im Gesamt-Datenflussdiagramm identifizierten Assets entsprechende Bedrohungen basierend auf dem STRIDE-Modell zugeordnet und diese zu einer gemeinsamen Bedrohungsliste zusammengeführt. Außerdem wurden die Bedrohungen mit der DREAD-Methode (Damage, Reproducibility, Exploitability, Affected Users, Discoverability) bewertet, um bei der späteren Festlegung der Spezifikationen des lokalen Marktes schwerwiegenden Sicherheitsrisiken entsprechend priorisieren zu können. Der Fokus der HSB bestand darin, als AP-Leiter die im Arbeitspaket anstehenden Aufgaben zu koordinieren und bei den Projektpartnern ein allgemeines Verständnis für das Thema IT-Sicherheit in der Blockchain und die dort vorhandenen Angriffsvektoren zu schaffen. Es wurden die Gemeinsamkeiten verschiedener Blockchain-Frameworks hinsichtlich ihrer Angriffsvektoren betrachtet und mit bestehenden Empfehlungen des BSI abgeglichen, die ebenfalls mit in das am Ende des Arbeitspaketes erstellte Bedrohungsanalyse-Dokument eingeflossen sind. Das Arbeitspaket „AP 2.2 - Bedrohungsanalyse“ konnte somit ohne Verzögerungen abgeschlossen werden.

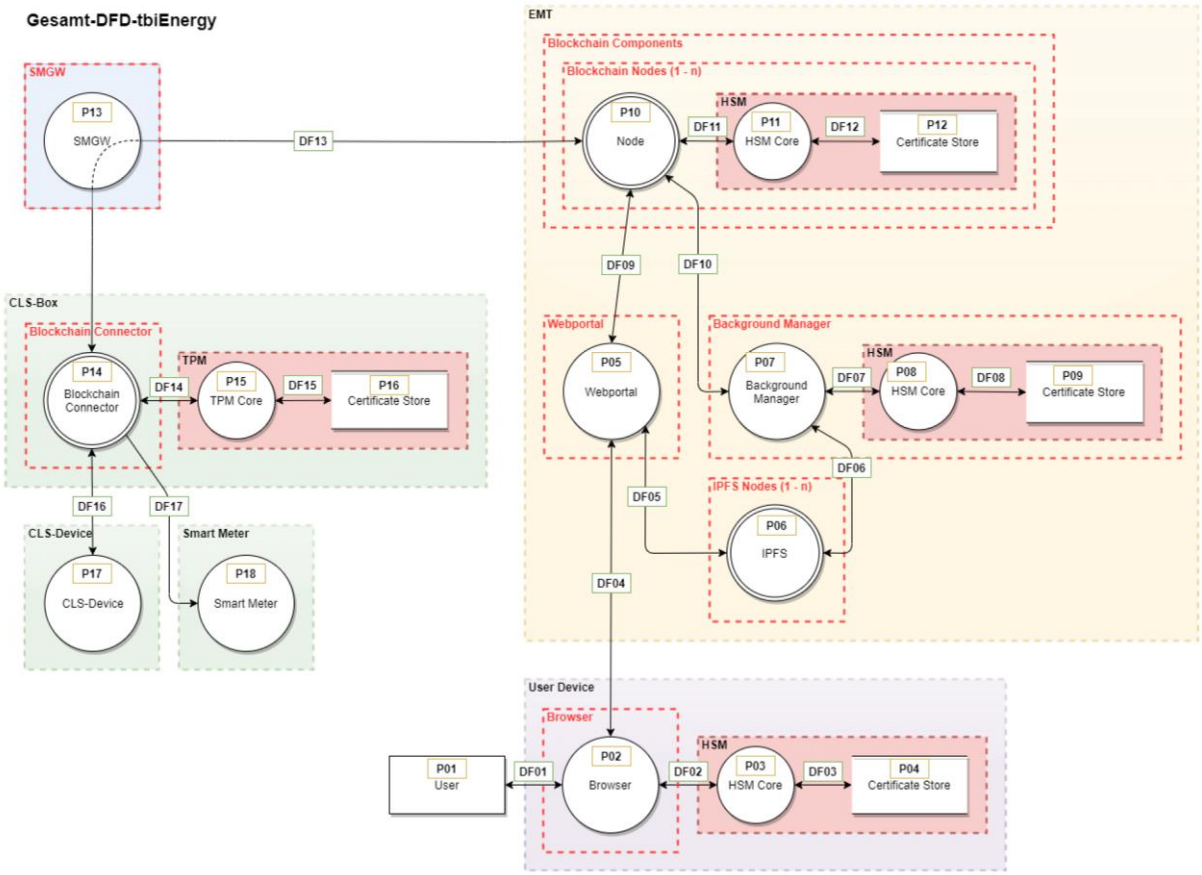

Abbildung 4: Gesamt-Datenflussdiagramm

3 Erstellen eines Smart-Contract-Konzeptes aus den Use Cases

Die HSB hat als Leiter des Arbeitspaketes die Koordination der Tätigkeiten und Organisation notwendiger Arbeitstreffen übernommen.

3.1 User Stories und Lastenhefte

Um die Use Cases in für Smart Contracts anwendbare Komponenten, Prozesse und Aufgaben überführen zu können, wurden diese im ersten Schritt gemeinsam mit den Projektpartnern in Form von User Stories erfasst. Im Anschluss wurden die User Stories in die folgend aufgeführten Lastenhefte überführt:

- Lastenheft_01_lokaler-Energiemarkt
- Lastenheft_02_Bilanzkreispool
- Lastenheft_03_Power2XToken
- Lastenheft_04_P2P-Handel
- Lastenheft_00_Hardwaresicherheit

3.2 Abstrakte technologieneutrale Beschreibung der Smart Contracts

Eine weitere Aufgabe im Rahmen des Arbeitspakets war die abstrakte Beschreibung der Inhalte der Smart Contracts mit Hilfe einer technologieneutralen Beschreibungssprache. Im ersten Schritt wurden durch die HSB mögliche Ansätze zur technologieneutralen Beschreibung von Smart Contracts evaluiert und die Smart Contract Description Language (SCDL) ausgewählt. In Vorbereitung auf die Erfassung der Beschreibungen der Smart Contracts wurde ein Validierungsschema in Form eines JSON-Schemas erstellt und in einen JSON-Editor integriert.

Diese Tools wurden den Projektpartnern für die anschließende Erfassung der Inhalte der Smart Contracts bereitgestellt. Die Erfassung der per SCDL beschriebenen Smart Contracts wurde im ersten Schritt durch die jeweils zuständigen Projektpartner vorangetrieben und zusätzlich in gemeinsamen Arbeitstreffen harmonisiert und finalisiert. Dabei wurden die folgenden abstrakt beschriebenen Smart Contracts erfasst:

- SC_tbi_Bilanzkreispool
- SC_tbi_CLS_Box
- SC_tbi_lokaler_Energiemarkt
- SC_tbi_token

3.3 Auswahl geeigneter Konsensalgorithmen

Um geeignete Konsensalgorithmen für den Einsatz im Rahmen des Projekts zu identifizieren, wurden in Zusammenarbeit mit dem Projektpartner Arxum die folgenden an den Konsensalgorithmus gestellten Anforderungen erarbeitet:

- Energieeffizienz – Der Konsensalgorithmus muss energieeffizient sein
- Transaktions-Durchsatz – Der Konsensalgorithmus muss eine hohe Anzahl an Transaktionen von einer Vielzahl von Entitäten abwickeln können
- Dezentralisierung – Der Konsensalgorithmus sollte ein möglichst hohes Maß an Dezentralität ermöglichen
- Skalierbarkeit – Der Konsensalgorithmus sollte gut skaliert werden können und dabei ebenfalls keinen großen Overhead verursachen
- Node-Identität – Die Identitäten der Betreiber der Full-Nodes sollten bekannt sein
- Ausführbarkeit von Code – Für die Umsetzung des lokalen Energiemarktplatzes ist die Ausführbarkeit von Code (Smart Contracts) zwingend notwendig

Dieser Anforderungskatalog wurde dann anschließend mit den Eigenschaften verschiedener Konsensalgorithmen abgeglichen. Dabei wurden die folgenden zwei für den Einsatz infrage kommende Konsensalgorithmen identifiziert:

- Delegated Proof of Stake (DPoS)
- Tendermint

3.4 Weiterentwicklung des Konzepts für den lokalen Energiemarktplatz

Um eine Grundlage für die abstrakt beschriebenen Smart Contracts zu erhalten, wurde das Konzept des lokalen Energiemarkts weiter spezifiziert und in eine objektorientierte Struktur überführt. In Abstimmung mit dem Projektpartner SWT wurde außerdem das durch die HSB konzipierte Zeitslotmodell für die Vermarktung der Energie weiterentwickelt und gemeinsam mit den Projektpartnern im Rahmen von Arbeitstreffen harmonisiert.

4 Anforderungs- und Schnittstellendefinition

Die HSB hat als Leiter des Arbeitspakets die Koordination der Tätigkeiten und Organisation notwendiger Arbeitstreffen übernommen. Es wurde zusammen mit den Projektpartnern eine ganzheitliche Architektur erarbeitet, die alle logischen und physischen Schnittstellen beinhaltet. Überdies wurden Auswahlkriterien für die auszuwählende Blockchain-Technologie definiert und Proof of Concepts für kritische Schnittstellen und Komponenten durchgeführt.

4.1 Entwurf der Architektur des Gesamtsystems

Im ersten Schritt wurde ein Entwurf der Architektur des Gesamtsystems angefertigt, wobei die IT-Sicherheitsanforderungen anhand der Erkenntnisse aus den vorhergehenden APs und Schutzprofilen für intelligente Messsysteme des BSI herangezogen wurden. Im Rahmen gemeinsamer Arbeitstreffen wurde der Entwurf iterativ weiterentwickelt und die kritischen Schnittstellen und Komponenten identifiziert.

4.2 Proof of Concepts

Für alle als kritisch erfassten Schnittstellen und Komponenten wurden die folgend aufgeführten Proof of Concepts (PoCs) bearbeitet.

- PoC_01 – Proxy Re-Encryption
- PoC_02 – openEMS
- PoC_03 – TPM/HSM Integration in Docker (virtualized TPM)
- PoC_04 – AWS HSM Integration
- PoC_05 – Software TPM, Integration in Docker
- PoC_06 – Evaluierung Smartcard Applets / UserHSM
- PoC_07 – Evaluierung Anbindung SmartMeter Lesekopf_info_dss
- PoC_08 – MaStR-Connector
- PoC_09 – Schnittstelle Prognosedaten

Im Rahmen der PoCs wurden die Aufgabenfelder jeweils durch einen oder mehrere Projektpartner bearbeitet und bei Bedarf gemeinsame Arbeitstreffen durchgeführt.

4.3 Auswahlkriterien für die Blockchain-Technologie

In Zusammenarbeit mit den Projektpartnern wurden Auswahlkriterien für die in AP 3.2 ausgewählten geeignete Blockchain-Technologie zur späteren Auswahl einer geeigneten Blockchain-Technologie erarbeitet (siehe Tabelle 1).

Tabelle 1: Erarbeitete Auswahlkriterien

Kriterium	Beschreibung
Konsensalgorithmus	DPOS, Tendermint (siehe AP2.3 - Auswahl der Konsensalgorithmen)
Interoperabilität	Interoperabilität sollte grundsätzlich langfristig möglich sein, wird aber im Rahmen des Projekts nicht umgesetzt.
Reifegraf/Entwicklungs-aktivität	Recherche zum Reifegrad der Blockchain-Technologie und Bewertung der aktuellen Entwicklungsaktivität.
Dezentralität	Die einzusetzende Blockchain-Technologie sollte eine dezentrale Verteilung der Komponenten in einer Konsortialen-Blockchain ermöglichen.
Verschlüsselung von OnChain-Daten (private Transaktionsdaten)	Eine Verschlüsselung der OnChain-Daten sollte entweder bereits durch die Blockchain-Technologie unterstützt werden oder zumindest sollte der Implementierung einer solchen Struktur nichts entgegenstehen.
Integrierbarkeit des HSMs der IFAG	Abstimmung mit der IFAG, inwiefern sich die durch die Blockchain-Technologie eingesetzten Signaturalgorithmen und Zertifikatstypen mit dem HSM der IFAG verbinden lassen.
Umsetzungsaufwand	Inwiefern entsprechen die Planungsgrößen den angesetzten Implementierungsaufwänden aus dem Antrag.

5 Blockchain-Technologie

Das Teil-Arbeitspaket AP 3.2 wurde durch den Projektpartner Infineon AG koordiniert und betreut. Die HSB hat die Projektpartner im Rahmen des Arbeitspakets bei der Auswahl einer geeigneten Blockchain-Technologie unterstützt. Hierzu wurden im ersten Schritt die im Kontext von AP2 erarbeiteten Auswahlkriterien für eine im Rahmen des Projekts geeignete Blockchain-Technologie betrachtet, welche die im Projekt verfolgten Schutzziele (Integrität, Verfügbarkeit, Vertraulichkeit, Authentizität und Anonymität) unterstützen. Zusammen mit dem Projektpartner Arxum wurden anwendbare Blockchain-Technologien identifiziert und verglichen. Hinsichtlich einsetzbarer Konsensalgorithmen wurden Blockchain-Technologien mit einbezogen, welche in die AP 2 identifizierten Konsensalgorithmen Delegated Proof of Stake (DPoS) oder Tendermint nutzen und somit unter anderem den für die Umsetzung des Demonstrators notwendigen Datendurchsatz gewährleisten können. Weitere Vergleichskriterien, die herangezogen wurden, waren die Integrierbarkeit des HSM (OPTIGA™ TPM) des Projektpartners Infineon AG, die Möglichkeit Daten OnChain verschlüsseln zu können (Proxy Re-Encryption) und der Reifegrad bzw. die Entwicklungsaktivität der Technologien, welche anhand des Fundamental Crypto Asset Score (FCAS) ermittelt wurden. Außerdem wurde der Aufwand für die Umsetzung des Demonstrators und die Möglichkeit der Interoperabilität mit anderen DLT-Systemen berücksichtigt. Eine Vorauswahl nutzbarer Blockchain-Technologien und ihre Bewertung hinsichtlich der zuvor aufgestellten Kriterien ist der Tabelle 2 zu entnehmen.

Tabelle 2: Vergleich Blockchain-Technologien

	EOS	Tendermint /Cosmos (ATOM)	TRON	Tezos	Lisk
Konsensalgorithmus	DPoS	Tendermint	DPoS	DPoS	DPoS
Integrierbarkeit des HSMs der IFAG (Digital Signature Algorithm)	ECDSA (secp256k1, **secp256r1**)	ECDSA (secp256k1, **secp256r1**)	ECDSA (secp256k1)	Schnorr (ed25519) ECDSA (secp256k1, **secp256r1**)	Schnorr (ed25519)
Verschlüsselung von OnChain-Daten (private Transaktionsdaten)	Proxy Re-Encryption (nuCypher)				
Reifegraf/Entwicklungsaktivität (FCAS)	892 A	794 A	877 A	737 B	744 B
Umsetzungsaufwand					
Interoperabilität					

Als Ergebnis des Arbeitspakets konnte mit EOS eine Blockchain-Technologie ausgewählt werden, die es ermöglicht, zur Steigerung der Sicherheit das HSM zu integrieren und die im Projekt priorisierten Schutzziele zu erfüllen. Aufgrund der Vorkenntnisse des Projektpartners ARXUM mit der Blockchain-Technologie EOS wurde im Konsortium gemeinsam beschlossen, das Kriterium des Umsetzungsaufwandes höher als das Kriterium der Interoperabilität zu gewichten. Diese Entscheidung stützt sich unter anderem auf das Interoperabilitätskriterium aus AP 2.4 das beschreibt, dass die Interoperabilität grundsätzlich langfristig möglich sein sollte, aber im Rahmen des Projekts nicht umgesetzt wird. Außerdem konnte mit dem Abschluss von AP 3 eine vollständige Beschreibung der für die Umsetzung des Demonstrators notwendigen Komponenten, ihrer Schnittstellen und der notwendigen Sicherheitsarchitektur erreicht werden, welche die Abbildung 5 zeigt. Der Meilenstein 1 konnte somit erreicht werden.

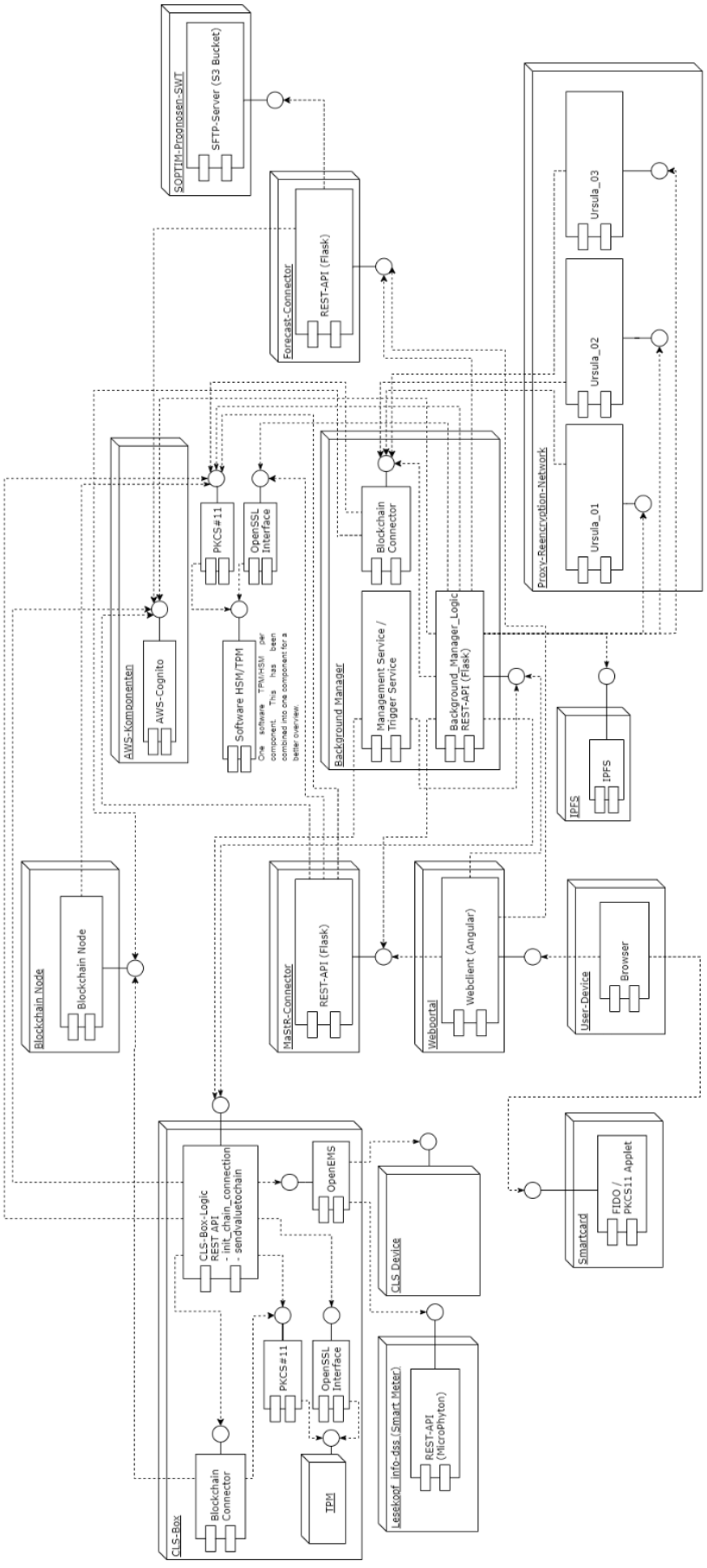

Abbildung 5: Komponentendiagramm Demonstrator

6 Blockchain Softwarebasis

Die Arbeiten an AP 4.1 werden nachfolgend in Unterarbeitspaketen aufgeschlüsselt beschrieben.

6.1 Etablierung eines agilen Entwicklungsrahmens

In Zusammenarbeit mit allen an der Softwareentwicklung beteiligten Projektpartnern wurde festgelegt, welche Tools, Frameworks und Programmiersprachen im Rahmen der Entwicklung des Demonstrators verwendet werden sollen. Es wurde durch den Projektpartner Arxum ein gemeinsamer GitLab-Server bereitgestellt. Die HSB richtete in ihrem internen Netz zudem einen lokalen Server als Entwicklungsinfrastruktur ein. Auf dieser lokalen Entwicklungsinfrastruktur wurde eine interne CI-Umgebung basierend auf GitLab aufgebaut, ein GitLab-Runner für den gemeinsamen GitLab-Server eingerichtet und angebunden. Die HSB hat außerdem einen CI-Prozess erarbeitet, mit dessen Hilfe die später umzusetzenden Komponenten des Demonstrators innerhalb von CI-Pipelines gebaut, mit Hilfe der Container-Technologie Docker gekapselt und in einer auf Amazon Web Services (AWS) basierenden Docker-Registry veröffentlicht werden können.

6.2 Entwurf des Software-Funktionsrahmens

In Vorbereitung auf AP 4.1.3 wurde gemeinsam mit den Projektpartnern ein umzusetzendes Szenario definiert, das den Funktionsumfang des umzusetzenden Demonstrators beschreibt. Die Softwareentwürfe der Komponenten der HSB wurden hierbei so konzipiert, dass im Rahmen des Feldtests ein technischer Durchstich erzielt werden kann. Hierzu wurden Funktionscluster identifiziert, in Form von Sequenzdiagrammen aufbereitet und feingranulare Softwareentwürfe für die spätere Aufschlüsselung in Form von Tickets im Zuge der agilen Entwicklung festgehalten. Im Folgenden werden die durch die HSB entworfenen Komponenten des Demonstrators vorgestellt.

6.2.1 MASTR-CONNECTOR

Um die Daten der Anlagenbetreiber auf dem zu entwickelnden lokalen Energiemarkt nicht vollständig neu erfassen zu müssen, hat die HSB einen Connector zum Markstammdatenregister (MaStR) entworfen. Im durch die Bundesnetzagentur betriebenen MaStR müssen Strom- und Gaserzeugungsanlagen verpflichtend registriert werden und können eindeutig identifizierbaren Marktakteuren zugeordnet werden. Das MaStR verfügt über eine SOAP-Webschnittstelle, die aber keine Filterung aller Einheiten hinsichtlich der Marktakteure zulässt. Der MaStR-Connector wurde daher als REST-Webservice konzipiert, der eine mit dem MaStR synchronisierte Kopie der MaStR-Einheiten verwaltet und gleichzeitig eine flexiblere Möglichkeit bietet, diese für tbiEnergy aufbereitet zu verarbeiten und bereitzustellen. Die Funktionalität des MaStR-Connectors umfasst die Synchronisierung über die Webschnittstelle des MaStR, die Abfrage von Details zu Einheiten anhand ihrer MaStR-Nr, die Abfrage aller Einheiten eines Marktakteures und die Evaluierung einer geeigneten Möglichkeit für das Deployment in die im Rahmen des Projekts genutzten AWS-Cloud. Die Funktionen wurden mit Hilfe der OpenAPI-Spezifikation erfasst und dem REST-API-Dokumentationstool Swagger UI visualisiert, wie in Abbildung 6 zu sehen ist.

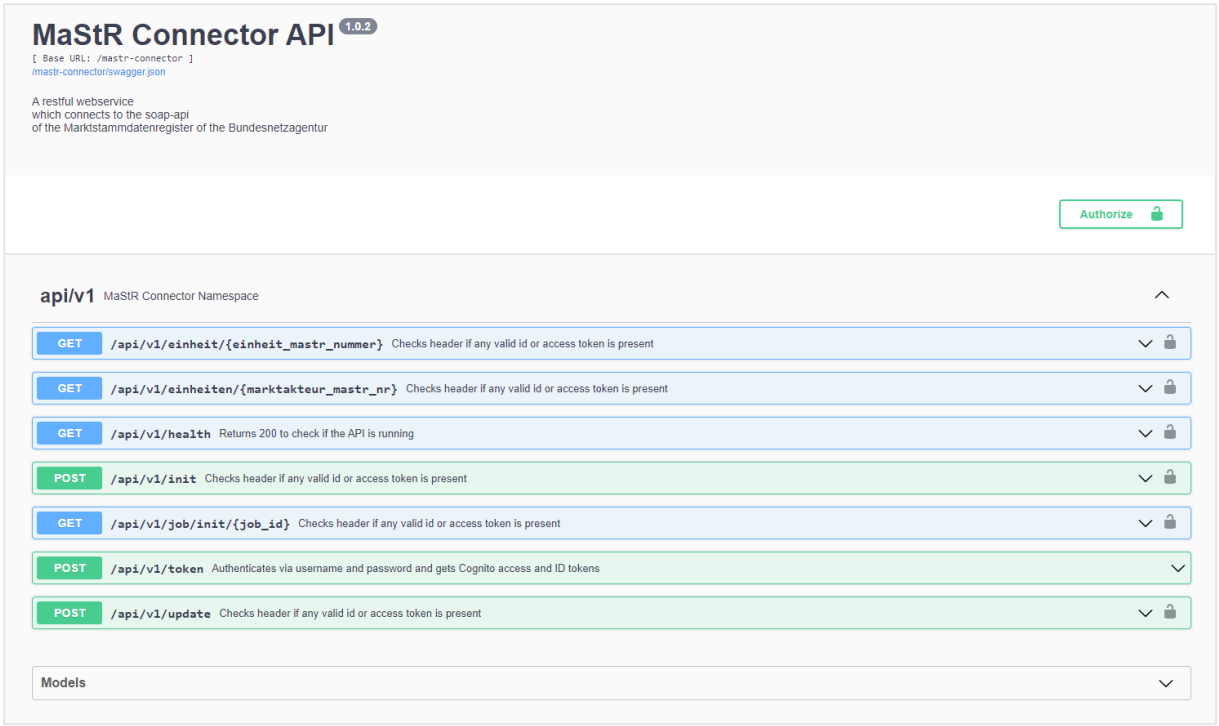

Abbildung 6: MaStR-Connector API - Visualisierung

6.2.2 FORECAST-CONNECTOR

Um den Bilanzkreis des lokalen Energiemarkts gegen Manipulationen abzusichern und Prognosen für die voraussichtliche Produktion oder den Verbrauch von Einheiten zu erhalten, wurde in Abstimmung mit den Stadtwerken Trier (SWT) das Konzept für die Entwicklung des Forecast-Connectors erstellt. Der Forecast-Connector soll zum einen Prognosen für Verbraucher wie Haushaltskunden, Wärmepumpen, Elektroautos und andere Verbrauchseinheiten aus dem MaStR bereitstellen. Zum anderen soll er die Menge der durch Erzeugungseinheiten wie z. B. Photovoltaikanlagen oder anderer im MaStR erfasster Anlagen und die durch sie voraussichtlich zur erwartenden Erzeugungsmenge prognostizieren. Der Forecast-Connector wurde als REST-Webservice konzipiert und soll die bereits beim Projektpartner SWT erstellten Erzeugungs- und Verbrauchsprognosen verarbeiten und für den Demonstrator bereitstellen. Um den Prozess der Bereitstellung und der Verarbeitung der Daten durch den Forecast-Connector mit dem Projektpartner SWT abzustimmen, wurden die Prozesse in Form von Sequenzdiagrammen (siehe Abbildung 7 und Abbildung 8) erfasst.

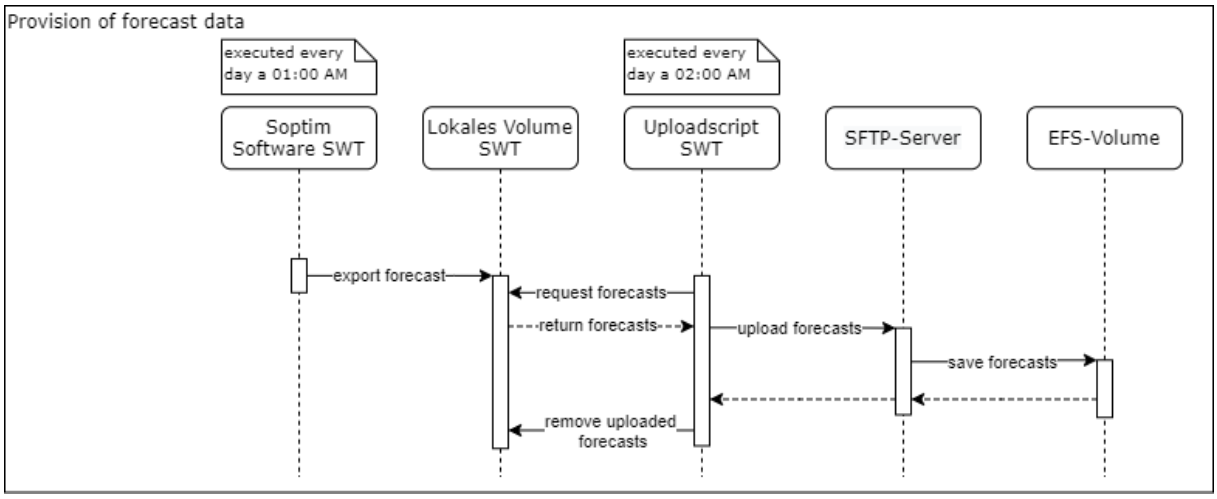

Abbildung 7: Sequenzdiagramm - Bereitstellung der Prognosedaten für den Forecast-Connector

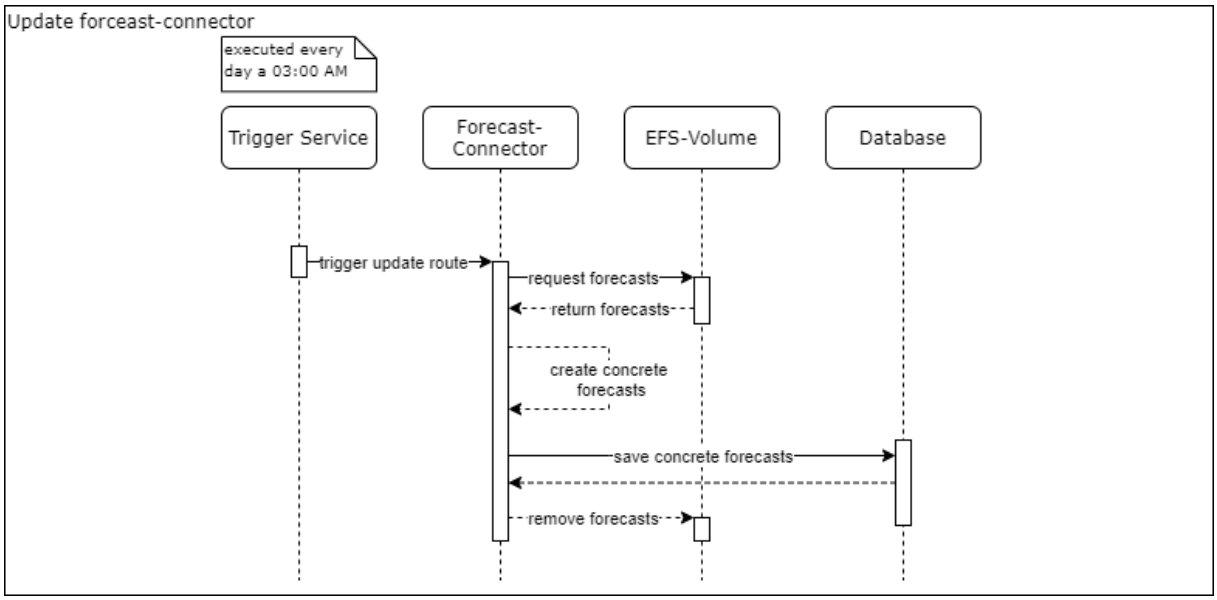

Abbildung 8: Sequenzdiagramm - Verarbeitung der Prognosedaten im Forecast-Connector

6.2.3 CLS-BOX-LOGIC

Die CLS-Box-Logic ist ein RESTful-Webservice der als Steuerkomponente der CLS-Box konzipiert ist, auf den CLS-Boxen ausgeführt wird und das Bindeglied zwischen der jeweiligen CLS-Box und der Microservice-Architektur in der AWS-Cloud bildet. Alle Teilnehmer, die am lokalen Energiemarkt partizipieren möchten, erhalten im Rahmen des Feldversuchs eine CLS-Box des Projekt-Partners devolo. Mithilfe der CLS-Box-Logic lässt sich der Zustand der CLS-Box abfragen und es können Daten von angeschlossenen Erzeugungsanlagen, Verbrauchsanlagen, Speicheranlagen und Grid Metern indirekt über OpenEMS[3], einem open-source Energiemanagementsystem, abgerufen werden. Außerdem werden Infrastrukturdienste in Form von Docker-Containern auf der CLS-Box verwaltet. Die CLS-Box-Logic stellt dabei sicher, dass physikalisch angebundene CLS-Boxen und die mit ihnen verbundenen Geräte vor der logischen Anbindung an den lokalen Energiemarkt validiert werden. Des Weiteren wurden Funktionen zur Steuerung von Datenerfassungs- und Steuerprozessen in Zusammenspiel mit den Diensten der Projektpartner integriert.

[3] URL: https://openems.io/

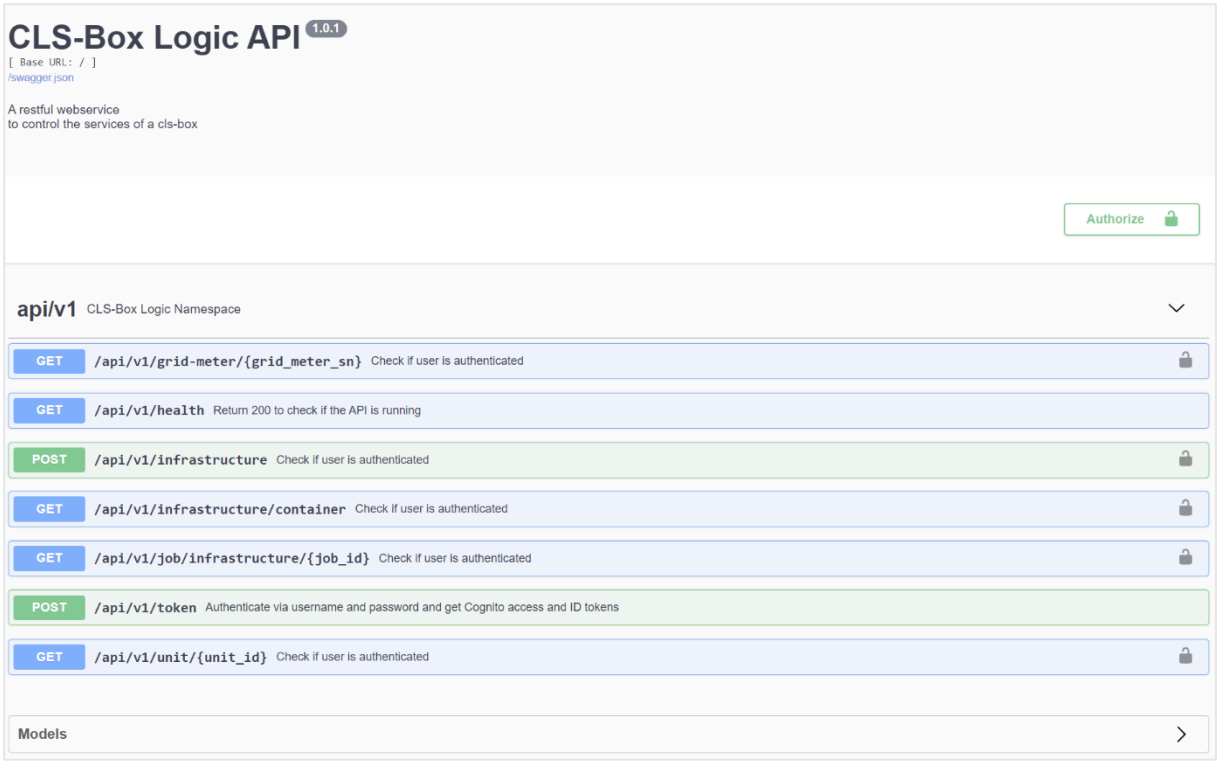

Abbildung 9: CLS-Box-Logic API – Visualisierung

6.2.4 TRIGGER-SERVICE

Um intern Funktionen der Systemkomponenten nach bestimmten Regeln auszulösen zu können, wurde ein Trigger-Service entworfen. Dieser ermöglicht das Auslösen der Synchronisierungsoperationen des MaStR-Connectors und die Berechnung der Prognosen des Forecast-Connectors. Ebenfalls können Events mit Hilfe des Aufrufs, der durch die Projektpartner bereitgestellten Smart Contracts ausgelöst werden.

6.2.5 WEBPORTAL

Das Webportal ist die Schnittstelle zum lokalen Energiemarkt für Verbraucher und Anlagenbetreiber, den Betreiber des Energiemarktes (EVU) sowie für Administratoren. Im Rahmen der Authentifizierung der Nutzer ist der Einsatz eines Nutzer-Hardware-Security-Moduls (User HSM) des Projektpartners Infineon Technologies AG vorgesehen. Der Benutzerrolle entsprechend werden nach erfolgter Authentifizierung verschiedene Funktionalitäten angeboten, die im Hintergrund die Backends der einzelnen Microservices aufrufen. Die Beschreibung der einzelnen Funktionen erfolgt daher separat je Microservice. Zusammenfassend kann das Webportal als Schnittstelle des Nutzers zum lokalen Energiemarkt betrachtet werden. Zusätzlich verfügt es über Funktionen, um die Hardwarekomponenten auf logischer Ebene anzubinden und zu verwalten (siehe Abbildung 10 bis 13).

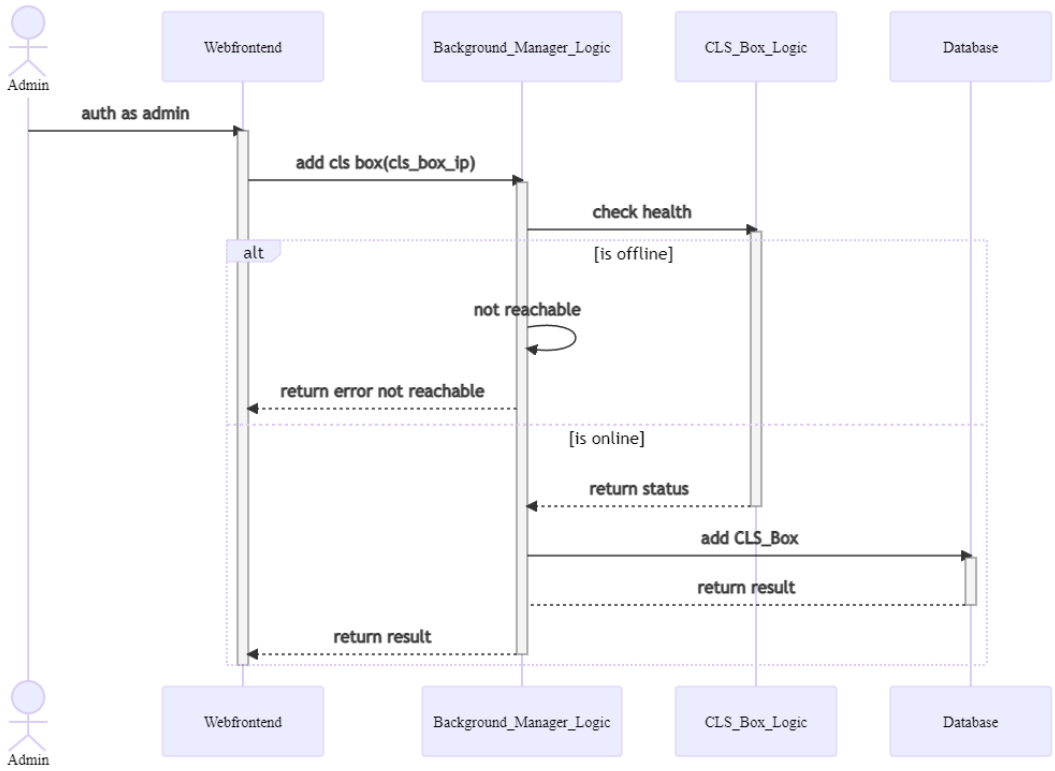

Abbildung 10: Sequenzdiagramm – Hinzufügen einer CLS-Box

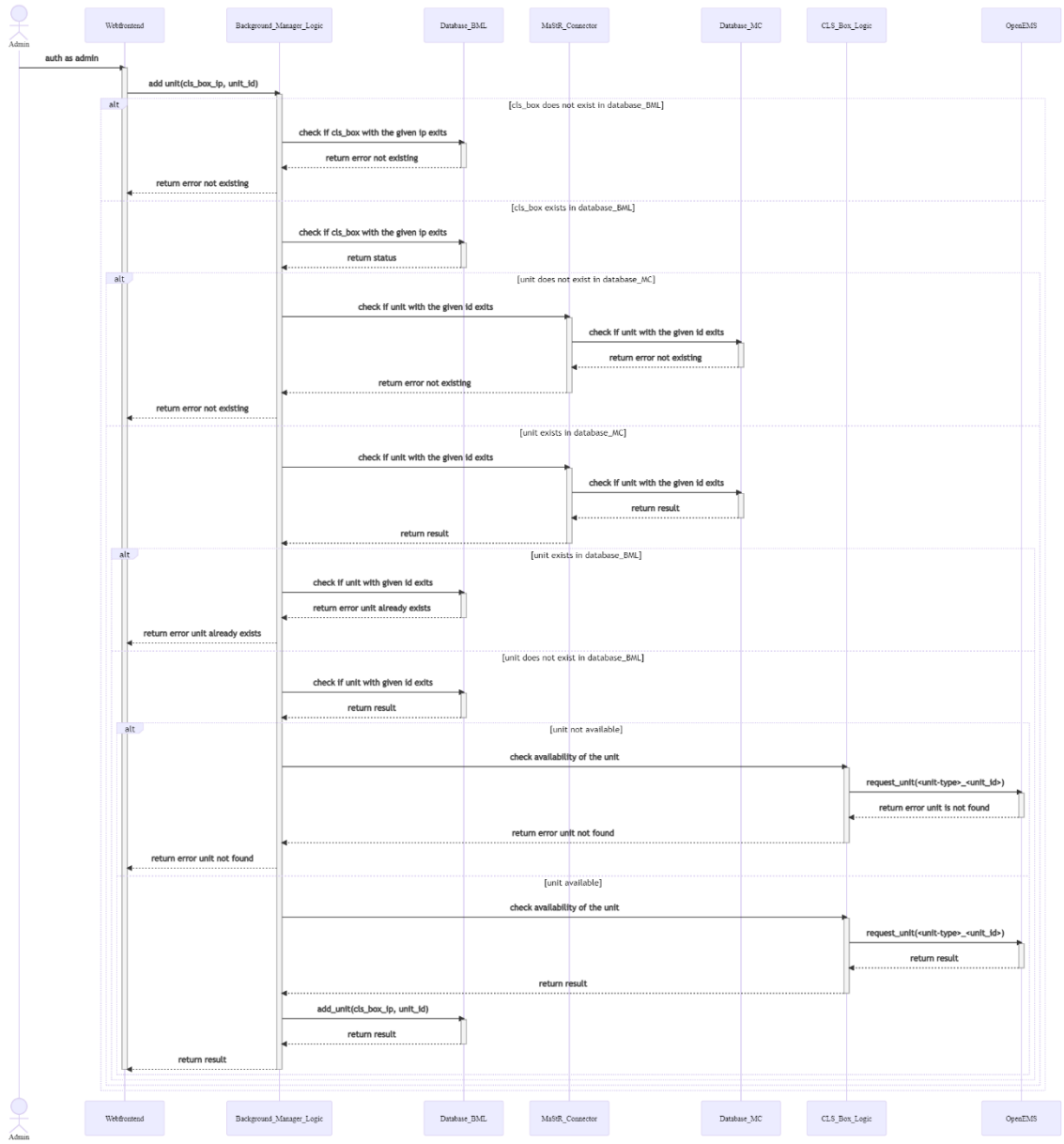

Abbildung 11: Sequenzdiagramm – Hinzufügen einer Einheit

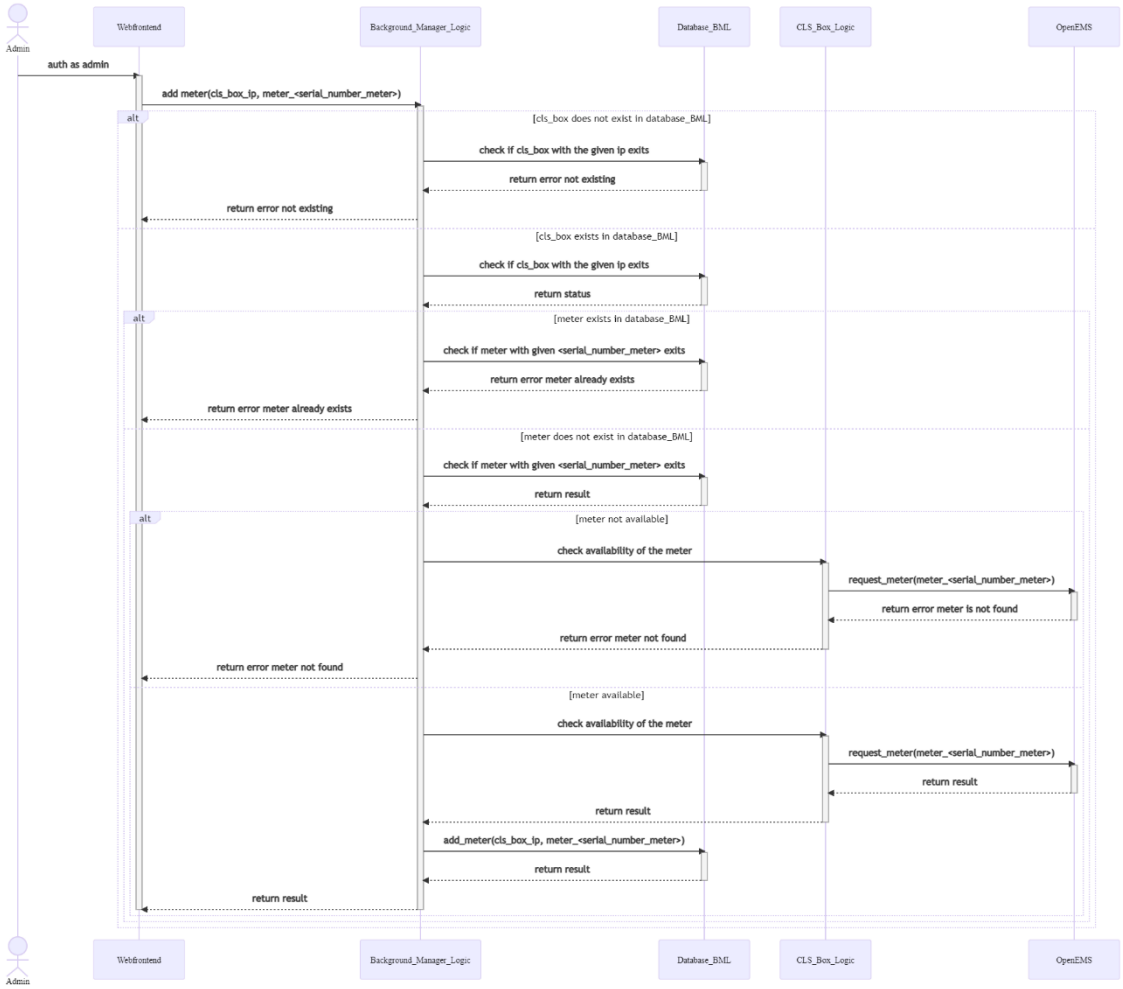

Abbildung 12: Sequenzdiagramm – Hinzufügen eines Meters

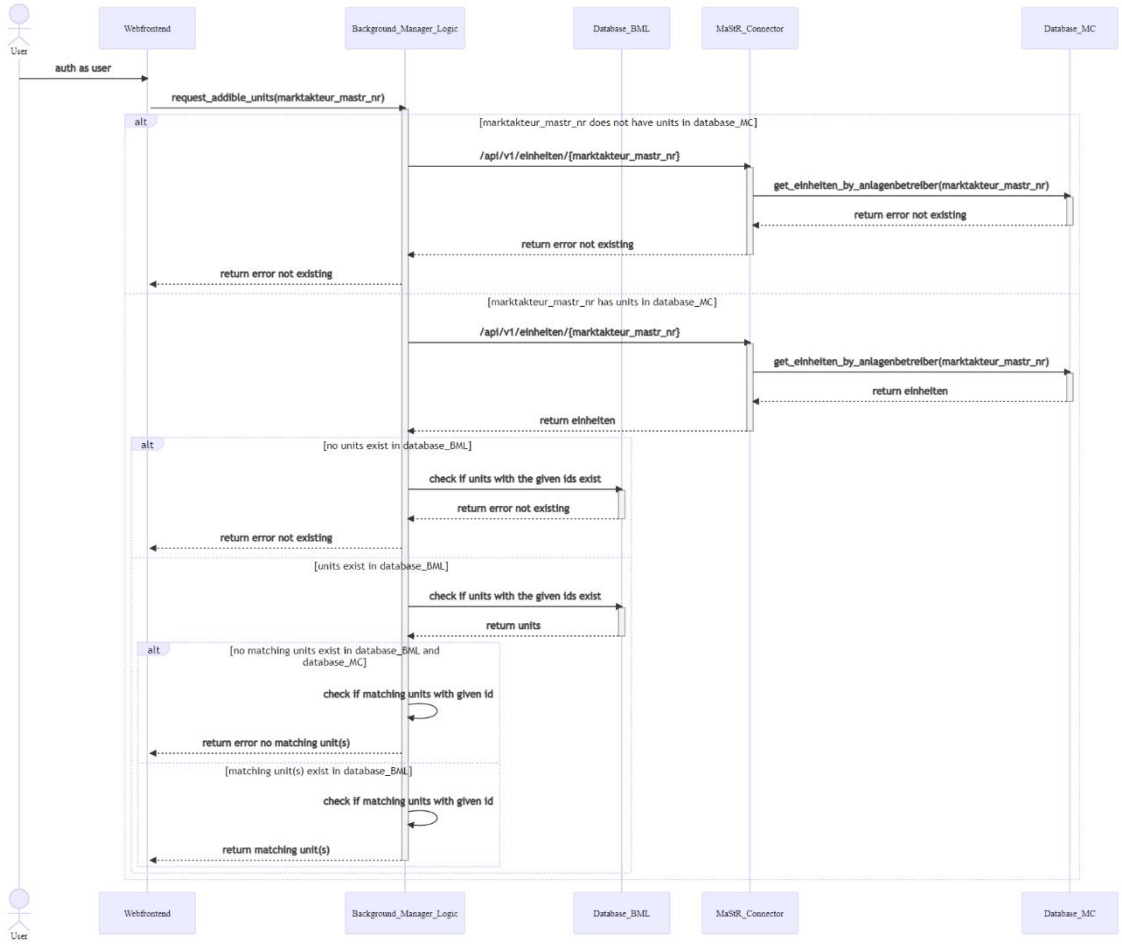

Abbildung 13: Sequenzdiagramm – Abruf hinzufügbarer Einheiten

6.2.6 BACKGROUND-MANAGER-LOGIC

Die Background-Manager-Logic stellt einen RESTful-Webservice innerhalb der Microservice-Architektur dar und befindet sich ebenfalls innerhalb der AWS-Cloud. Verwendet wird die Background-Manager-Logic durch das Webportal mit Hilfe eines erzeugten Angular-Clients. Die Background-Manager-Logic verfügt über eine Datenbankanbindung und stellt die in Abbildung 14 dargestellten Funktionalitäten zur Verfügung, die im Folgenden beschrieben werden.

Die Basisfunktionalitäten für administrative Benutzerrollen lassen sich wie folgt zusammenfassen:

- Hinzufügen einer physikalisch angebundenen und erreichbaren CLS-Box zur Datenbank
- Abfragen aller CLS-Boxen oder spezifischer CLS-Boxen anhand ihrer IP-Adresse in der Datenbank
- Hinzufügen eines Meters zur Datenbank
- Abfragen aller angebundenen Meter oder spezifischen Meter anhand der Seriennummer aus der Datenbank
- Hinzufügen einer Einheit zur Datenbank
- Abfragen aller angebundenen Einheiten oder spezifische Einheiten anhand der Einheiten-MaStR-Nummer von der Datenbank
- Abruf des Status des Services

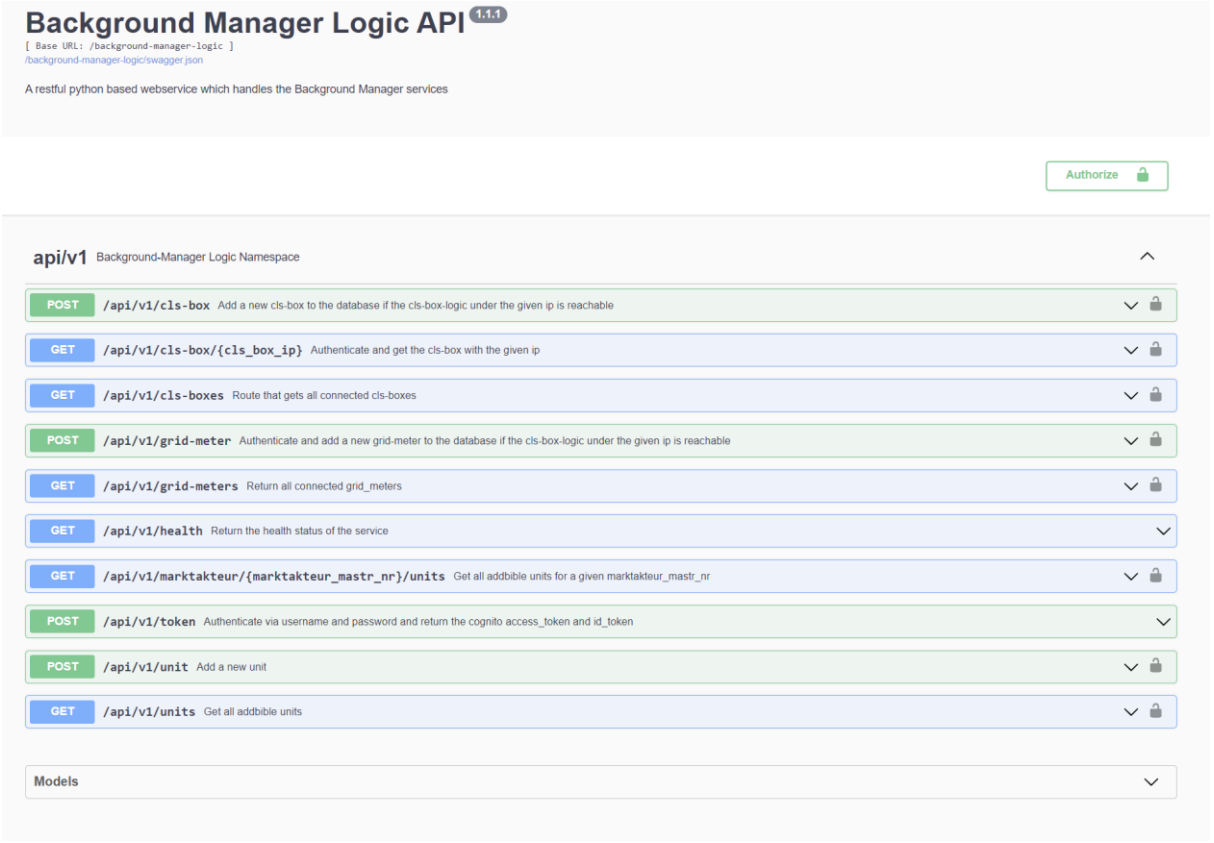

Abbildung 14: Background-Manager-Logic API – Visualisierung

Nutzer ohne erweiterte Nutzungsrechte können ihre eigenen Einheiten auf dem lokalen Energiemarkt registrieren. Hierzu wird eine von Arxum bereitgestellte API genutzt, mit der auf die Smart Contracts zugegriffen und auf die EOS-Blockchain geschrieben werden kann.

6.3 Testgetriebene Implementierung des Funktionsumfangs

Im Rahmen des Arbeitspaketes wurden die zuvor in AP 4.1.2 konzipierten Demonstrator-Komponenten fortlaufend gemeinsam mit den Projektpartnern spezifiziert und in ihrer Basisfunktionalität implementiert. Damit liegen alle gemeinsam mit den Projektpartnern definierten Teilkomponenten der HSB getestet und funktionsbereit vor. Der Meilenstein 2 wurde somit erreicht.

Im Folgenden wird der technische Aufbau der durch die HSB entwickelten REST-APIs erläutert, auf die technischen Besonderheiten der RESTful-Webservices eingegangen und das Webportal des lokalen Energiemarkts vorgestellt.

6.3.1 RESTful-WEBSERVICES – REST-APIS

Um eine Grundlage für die Entwicklung der Microservice-Architektur zu erarbeiten, wurde eine generische Projektstruktur für die RESTful-Webservices (REST-APIs) entwickelt (siehe Abbildung 15). Diese strukturiert die Webservices in die Bereiche RESTful Web Application, Testing, Deployment und Dokumentation.

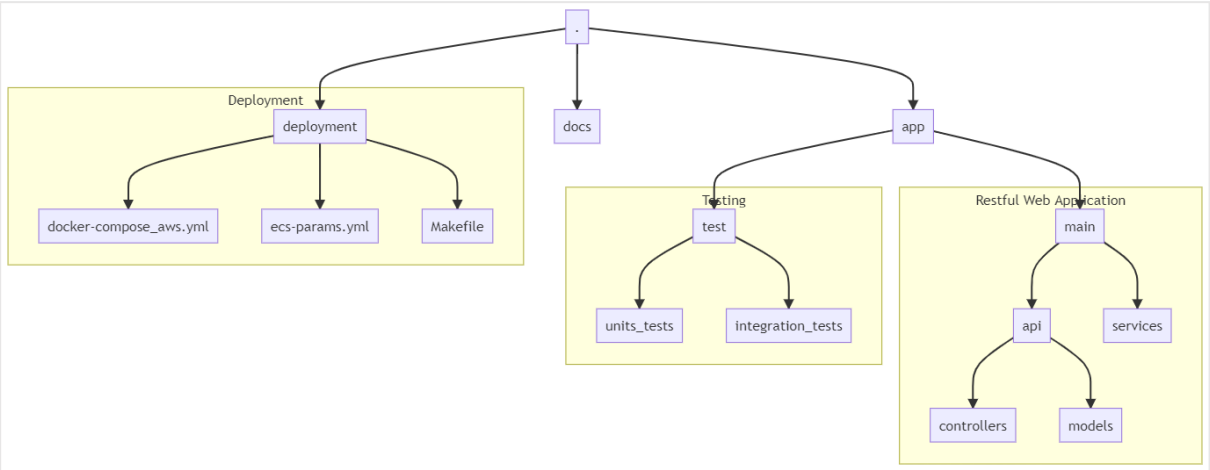

Abbildung 15: Projekt-Struktur RESTful-Webservices

Die RESTful-Webservices basieren auf dem Python-Framework Flask-RESTX, das eine Erweiterung des Flask-Frameworks darstellt und z. B. die Dokumentation der Schnittstellen mit Hilfe der sogenannten OpenAPI-Spezifikation integriert. Flask-RESTX generiert anhand von Annotationen ein OpenAPI-Dokument, welches eine standardisierte, sprachunabhängige Beschreibung der Funktionen der RESTful-Webservices enthält. Anhand des OpenAPI-Dokuments erfolgt außerdem die Generierung einer statischen Dokumentationswebseite, die Generierung der Clients (Python-Client sowie TypeScript-Client) und die Möglichkeit zur Darstellung der zur Verfügung stehenden Funktionen mit Hilfe von Swagger UI. Außerdem wurde ein generischer Authentifizierungs- und Autorisierungs-Service entwickelt, der den Nutzerverwaltungsdienst AWS Cognito integriert.

Um im Rahmen der prototypischen Entwicklung Fehler in den Webservices auszuschließen, wurde überdies entschieden, die Anwendungen testgetrieben zu entwickeln. Im Zuge der testgetriebenen Entwicklung erfolgte im ersten Schritt die Umsetzung der Unit- und Integrationstests und erst im Anschluss die Entwicklung der eigentlichen Funktionalitäten.

Da die Webservices im Rahmen der folgenden Arbeitspakete in die Cloud-Umgebung des Projektpartners Arxum integriert werden, wurden die Anwendungen mithilfe der Containerisierungslösung Docker in Containern gekapselt und somit eine Grundlage für das spätere Deployment in die Cloud-Umgebung geschaffen.

6.3.1.1 MASTR-CONNECTOR

Der MaStR-Connector ruft die Stammdaten (Strom- und Gaserzeugungsanlagen, Marktakteure wie Anlagenbetreiber, Netzbetreiber und Energielieferanten) über das Marktstammdatenregister ab und synchronisiert diese anschließend mit seiner eigenen Datenbank. Hierfür werden zwei Methoden genutzt:

a) Beim erstmaligen Befüllen der Datenbank durch Aufruf der „/init“-Route werden alle Stammdaten aus XML-Dateien geladen.

b) Durch das Aufrufen der „/update“-Route werden alle neuen Änderungen der Stammdaten seit der letzten Aktualisierung über die SOAP-API des MaStR abgerufen.

Im Anschluss werden die Stammdaten von XML in interne Datentypen geparst und in einer MongoDB-Datenbank gespeichert bzw. aktualisiert. Um etwaige Timeouts durch die großen Datenmengen zu vermeiden, werden diese Funktionen in einzelne Redis-Jobs ausgelagert. Der aktuelle Status der Jobs kann unter Angabe einer entsprechenden Job-ID abgefragt werden.

Der MaStR-Connector ermöglicht die Abfrage von Details der gespeicherten Einheiten anhand ihrer MaStR-Nr. Diese Details sind:

- MaStR-Nummer der Einheit
- Datum der letzten Aktualisierung
- Name der Einheit
- Einheittyp
- PLZ
- Ort
- Bruttoleistung
- Betriebsstatus
- MaStR-Nr. des Anlagenbetreibers

Überdies können alle Einheiten eines Marktakteures über den MaStR-Connector abgerufen werden.

6.3.1.2 FORECAST-CONNECTOR

Der Forecast-Connector ist eine der REST-APIs und stellt Operationen für Verbrauchs- und Erzeugungs-Prognosen über Routen bereit. Die Prognosen (Forecasts) werden für 15-minütige Zeitabschnitte (Timeslots) aufgestellt und von den SWT über ein EFS-Volume in der AWS-Cloud im .xlsx-Dateiformat bereitgestellt. Dabei werden neue Prognosen täglich hochgeladen und vom Forecast-Connector verarbeitet. Der Forecast-Connector liest die Prognosedaten aus und importiert die Forecasts und Timeslots in eine eigene Datenbank. Die importierten Timeslots und deren enthaltene Forecasts können via REST-Routen abgerufen werden, wobei die Ergebnisse mit den folgenden Kriterien gefiltert werden können:

- Zeitstempel (ISO 8601 Format: YYYY-MM-DDThh:mm)
- Zeitraum

Forecasts können auch unabhängig ihrer zugehörigen Timeslots abgerufen werden. Forecasts können wie folgt gefiltert werden:

- MaStR-Nummer der Einheit
- Anlagen-Typ (H0, E0, WP, PV, …)
- (Nach Timeslot-Zeitstempel für bestimmten) Zeitraum

6.3.1.3 BACKGROUND-MANAGER-LOGIC

Die Background-Manager-Logic bildet zusammen mit dem Triggerservice und dem Blockchain-Connector den Bereich des Background-Managers innerhalb der Systemarchitektur. Der Blockchain-Connector beinhaltet die Blockchain-Komponenten des Performers, Readers und Watchers. Neben der Ausführung von Hintergrundaufgaben stellt dieser Bereich das Bindeglied zum Blockchain-Netzwerk und der restlichen Microservice-Architektur dar. Während die Background-Manager-Logic die notwendigen Hilfsfunktionen bereitstellt, verbinden die Blockchain-Komponenten lokale und externe Datenquellen über die Background-Manager-Logic mit der Blockchain.

Um die Basisfunktionen für den administrativen Benutzer bereitzustellen, wird eine Amazon DocumentDB als Datenbankanbindung genutzt. Diese Datenbank mit MongoDB-Kompatibilität ist ein schneller, zuverlässiger und vollständig verwalteter Datenbankdienst der AWS-Cloud und bildet die Datengrundlage, um die gespeicherten energietechnischen Komponenten im Zuge der Integration von den Endnutzern am lokalen Energiemarkt registrieren zu können.

Innerhalb der agilen Arbeitsweise der HSB wurde auch für diesen Webservice eine kontinuierliche Automatisierung und Überwachung des Services sichergestellt. Hierzu wurden Aktualisierungen im

Code oder der Struktur vor dem Deployment durch die Nutzung von der CI/CD-Pipeline mit Hilfe von Integrations- und Unittests überprüft und anschließend ein aktualisiertes Docker-Image gebaut und auf die Amazon Elastic Container Registry (ECR) hochgeladen. Mit AWS Fargate wurde anschließend dieses neue Docker-Image aus der ECR als Container in einem Cluster des AWS Amazon Elastic Container Services (ECS) ausgeführt und bereitgestellt.

6.3.1.4 CLS-BOX-LOGIC

Die CLS-Box-Logic wird auf dem durch den Projektpartner devolo AG bereitgestellten ARMv7-Einplatinencomputer (CLS-Box) betrieben. Um die Lauffähigkeit der CLS-Box-Logic auf der CLS-Box zu gewährleisten, wurde die Erstellung eines Multi-Platform-Images mit Hilfe des Tools BuildX in die CI-Pipeline integriert. Somit kann nach der Fertigstellung eines neuen Features des Webservices durch die Angabe eines Tags automatisch die Erzeugung und der Upload des Images erfolgen.

Da die CLS-Box nur über eine begrenzte Speicherkapazität verfügt, wurden verschiede Maßnahmen unternommen, um den Speicherplatzbedarf des Containers zu minimieren. Als Basisimage für den Container wurde das minimalistische Docker-Image arm32v7/alpine verwendet. Außerdem wurden sowohl der Docker-Client als auch die Compose-Plugins des Hostsystems in den Container gemountet, um etwaige Redundanzen zu vermeiden. Als letzte Maßnahme wurden die Abhängigkeiten mit Hilfe des Tools PyInstaller analysiert und nur verwendete Abhängigkeiten inkludiert wodurch der Speicherplatzbedarf der Anwendung erheblich reduziert werden konnte.

Durch die beschränkte Rechenleistung der CLS-Box dauert die Ausführung einiger Prozesse relativ lange, daher wurden diese mit der Python-Bibliothek RQ in einen eigenen Worker-Container ausgelagert. Dieser ermöglicht es, lang andauernde Prozesse auszulagern und das Ergebnis zu einem späteren Zeitpunkt abzufragen.

6.3.2 TRIGGER-SERVICE

Der Trigger-Service ist als AWS-Cloud-Komponente implementiert und stellt eine Sammlung aus AWS-Lambda-Funktionen und AWS-EventBridge-Regeln dar.

AWS-Lambda-Funktionen sind Code, der als Reaktion auf Events ausgeführt wird. Die AWS-EventBridge-Regeln senden in festgelegten Zeitintervallen ebensolche Events an die Lambda-Funktionen, damit diese ausgeführt werden.

Die Lambda-Funktionen des Trigger-Services erfüllen folgende Aufgaben:

- Synchronisierung des MaStR-Connectors mit dem MaStR
- Synchronisierung des Forecast-Connectors mit den Prognosedaten der SWT
- Weiterleitung der Auslösung eines Events an die Systemkomponenten des Projektpartners Arxum

6.3.3 WEBPORTAL

Wie alle von der HSB entwickelten Komponenten wurde auch das Webportal (siehe Abbildung 16) agil entwickelt und unter Verwendung von CI/CD-Pipelines fortlaufend aktualisiert. Grundlage für das Webportal ist das Framework Angular. Es macht sich moderne Softwarekonzepte zu Nutze, um die Qualität der Software sicherzustellen. Zur Authentifizierung und Autorisierung wurde AWS Amplify in das Webportal integriert. Dieses Framework dient der Anbindung an die in Amazon Cognito hinterlegten Nutzerdaten. Dazu zählen sowohl die benötigten Informationen zur Authentifizierung als auch die zur Autorisierung.

Abbildung 16: Webportal

Zur Umsetzung der benötigten Funktionalität nutzt das Webportal die vier von der HSB entwickelten Microservices (Forecast-Connector, Background-Manager-Logic, MaStR-Connector und die Instanzen der CLS-Box-Logic auf den CLS-Boxen).

6.4 Integration der Funktionen zur Blockchain-Plattform

Die vorbereitenden Tätigkeiten für die Plattformintegration in AP 5.1 wurden gemeinsam mit den Projektpartnern durchgeführt. Die HSB hat dabei fortlaufend bei der Implementierung und der Integration der Systemkomponenten unterstützt. Dafür wurde die Basisfunktionalität der Systemkomponenten fortlaufend um die absehbar notwendigen Funktionen erweitert und die entwickelten Microservices der HSB für die Plattformintegration vorbereitet.

6.4.1 Registrierung einer Einheit auf dem lokalen Energiemarkt

Abbildung 17 bildet den Prozess der Registrierung einer Einheit auf dem lokalen Energiemarkt ab. Der Administrator kann über einen Dialog im Webportal (siehe Abbildung 18) eine zuvor in der Verwaltungsstruktur hinzugefügte Einheit auswählen und zum lokalen Energiemarkt (auf der Blockchain) hinzufügen. Durch die vorherigen Prozesse sind viele Vorbedingung schon überprüft worden und es kann sichergestellt werden, dass die aufgelisteten Einheiten bereits erfolgreich angebunden sind. Der Prozess überprüft dabei, ob die Einheit über die Softwarekomponenten erreichbar ist und fügt diese nach erfolgreicher Prüfung hinzu.

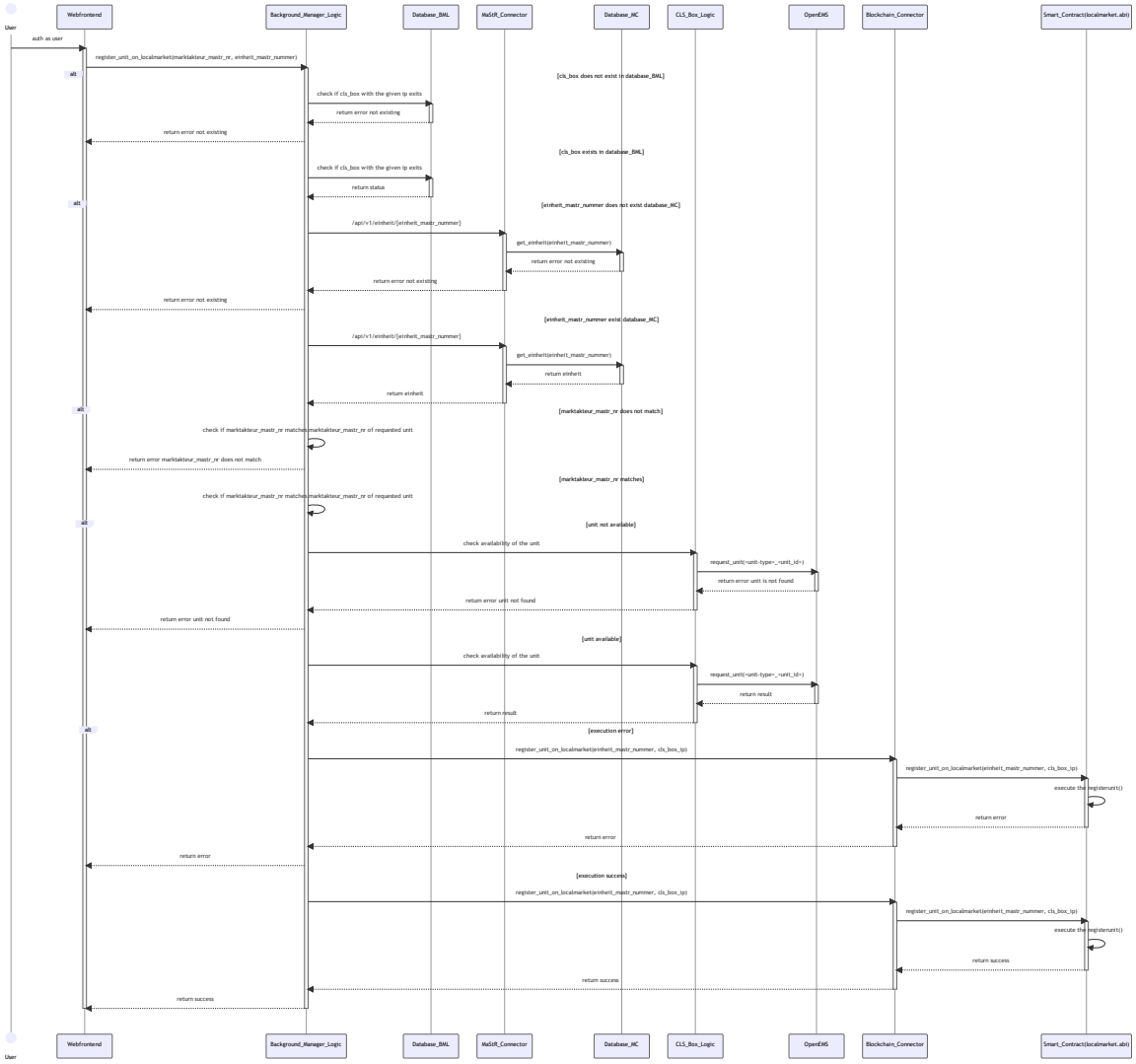

Abbildung 17: Sequenzdiagramm Registrierung einer Einheit

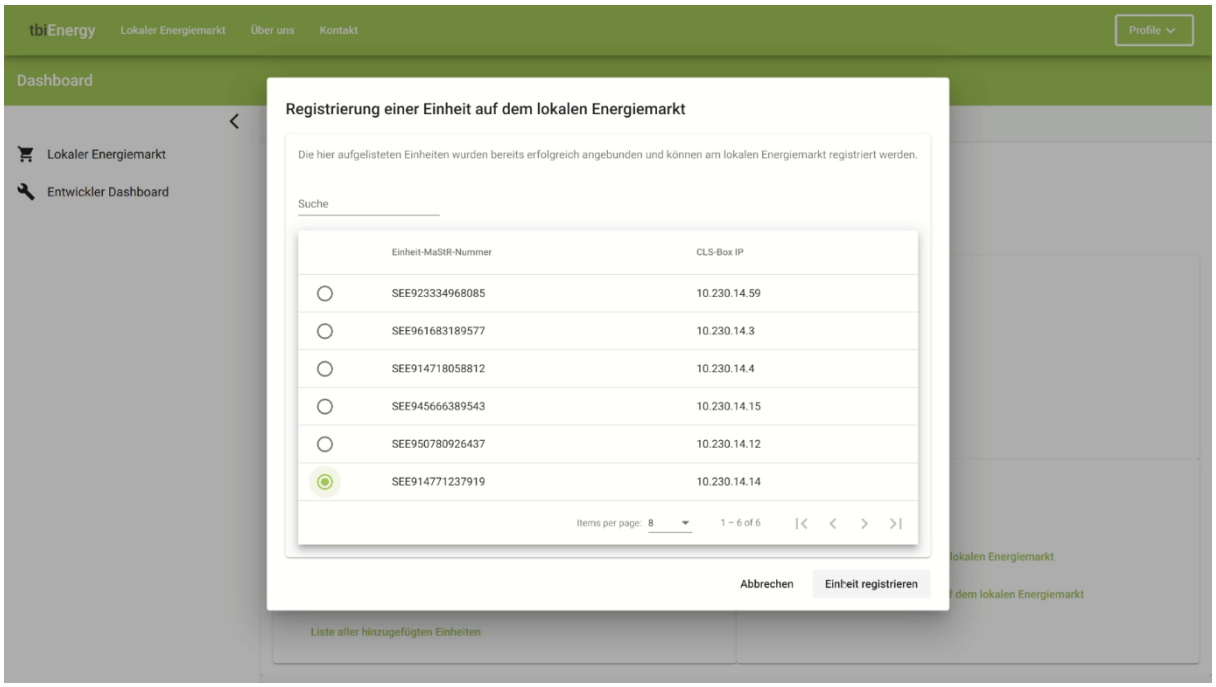

Abbildung 18: Registrierung einer Einheit (Webportal)

6.4.2 Einstellen von Angeboten auf dem lokalen Energiemarkt

Sobald Einheiten auf dem lokalen Energiemarkt hinzugefügt wurden, können Nutzer diese Anlagen anbieten (siehe Abbildung 19). Nutzer können hierfür im Webportal aus einer Liste der bereits für sie angemeldeten Einheiten eine Einheit auswählen und die weiteren Angaben wie den Angebotszeitpunkt, die Energiemenge und den gewünschten Startpreis angeben (vgl. Abbildung 20). Um zu vermeiden, dass Energiemengen angeboten werden, die nicht durch die Anlage geliefert werden können, wird die maximal anbietbare Energiemenge auf die zu erwartende Erzeugungsleistung abzüglich des zu erwartenden Verbrauchs des Haushalts begrenzt. Die Berechnung erfolgt durch den Forecast-Connector, der diese auf Basis der Prognosedaten des EVUs durchführt.

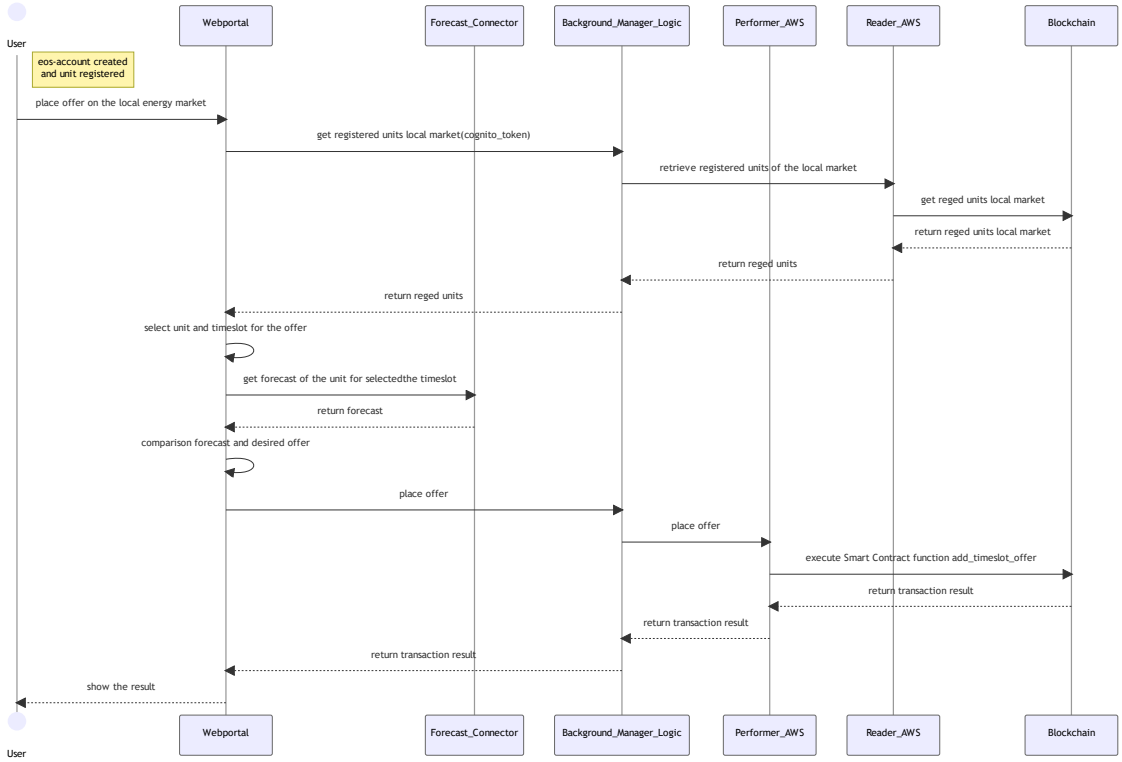

Abbildung 19: Sequenzdiagramm – Einstellen von Angeboten

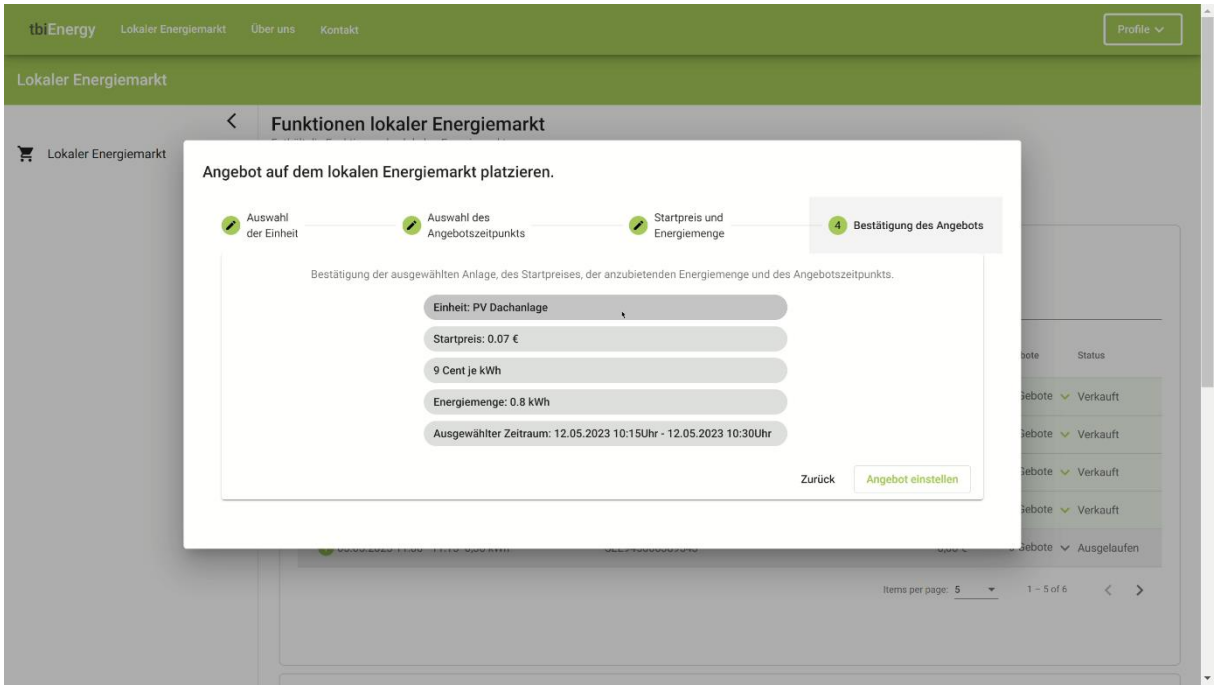

Abbildung 20: Einstellen von Angeboten (Webportal)

6.4.3 Bieten auf Angebote auf dem lokalen Energiemarkt

Nachdem Angebote auf dem lokalen Energiemarkt eingestellt wurden, erhalten andere Teilnehmer die Möglichkeit, Gebote auf diese Angebote abzugeben. Abbildung 21 beschreibt, wie dieser Prozess konzipiert wurde. Der Entwurf der notwendigen Serviceschnittstellen wurde in Abstimmung mit den

Projektpartnern erstellt und bei der Plattformintegration in AP 5.1 integriert. Auch dieser Prozess wurde, wie in Abbildung 22 zu sehen ist, über das Webportal in einem Dialog bereitgestellt.

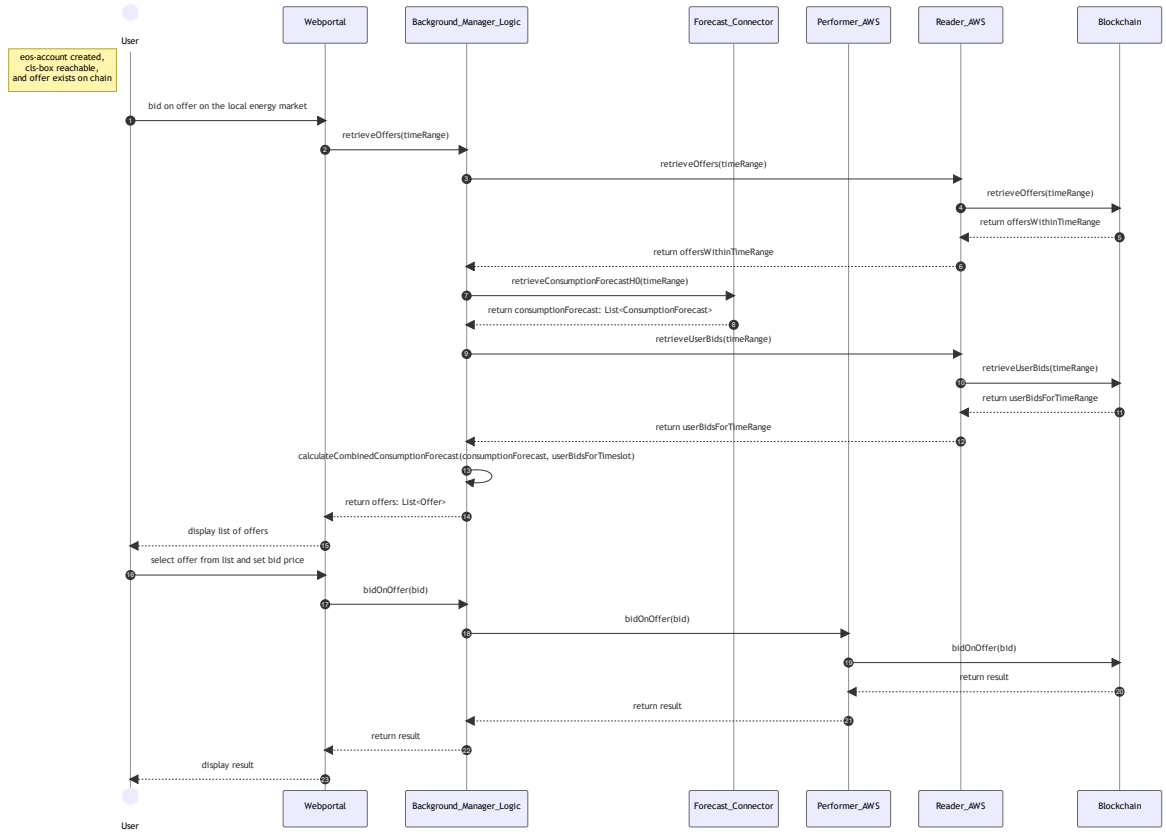

Abbildung 21: Sequenzdiagramm – Bieten auf Angebote

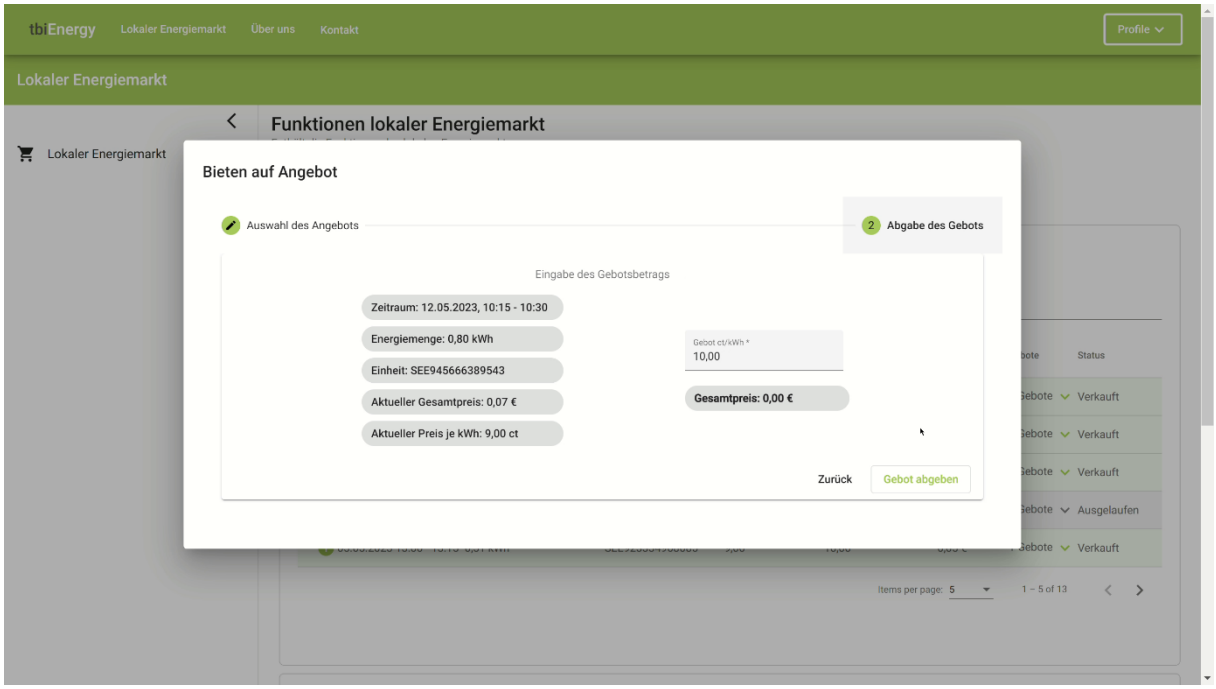

Abbildung 22: Bieten auf Angebote (Webportal)

6.4.4 Daten übertragen

Um beim erfolgreichen Verkauf eines Angebots die Erzeugungs- bzw. Verbrauchsdaten von einer CLS-Box in die Blockchain zu übermitteln, wurden in der Verarbeitungslogik der CLS-Box-Logic Funktionen zum Abruf und Versenden der Daten implementiert. Der Prozess orientiert sich an Abbildung 20 und wird durch ein Event auf der Blockchain ausgelöst. Anhand des Events entscheidet die CLS-Box-Logic, ob Daten für eine angebundene Einheit übermittelt werden müssen. Ist dies der Fall, ruft die CLS-Box-Logic die aggregierten Daten für den angeforderten Zeitraum von OpenEMS ab und übergibt sie an den CLS-Box-Performer, der die Daten anschließend durch die TPM API signieren lässt und die signierten Daten an die Blockchain überträgt. Die Übermittlung der Daten erfolgt hierbei sowohl von der Seite des Betreibers einer Anlage (Erzeugungsdaten) als auch von der Seite des Käufers (Verbrauchsdaten).

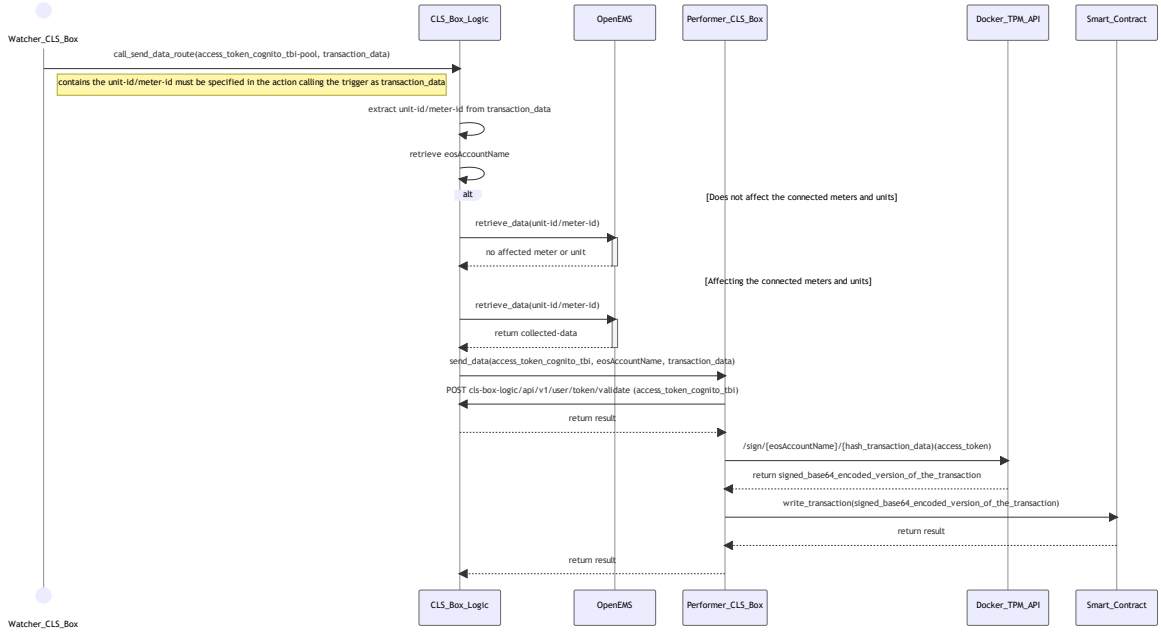

Abbildung 23: Sequenzdiagramm – Daten übertragen

6.4.5 Bilanz anzeigen

Im letzten Schritt werden die ermittelten Erzeugungsdaten und Verbrauchsdaten gegenübergestellt und die Bilanz je verkauftem Angebot berechnet, um Nutzern einen Überblick über die abgewickelten Prozesse zu geben. Diese Daten werden wie in Abbildung 24 beschrieben vom Backend angefordert und im Webportal durch entsprechende Diagramme aufbereitet (siehe Abbildung 25).

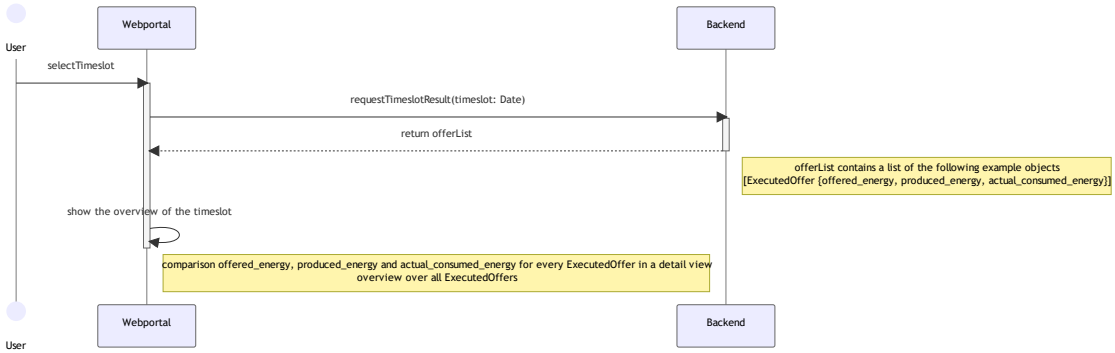

Abbildung 24: Sequenzdiagramm – Bilanz anzeigen

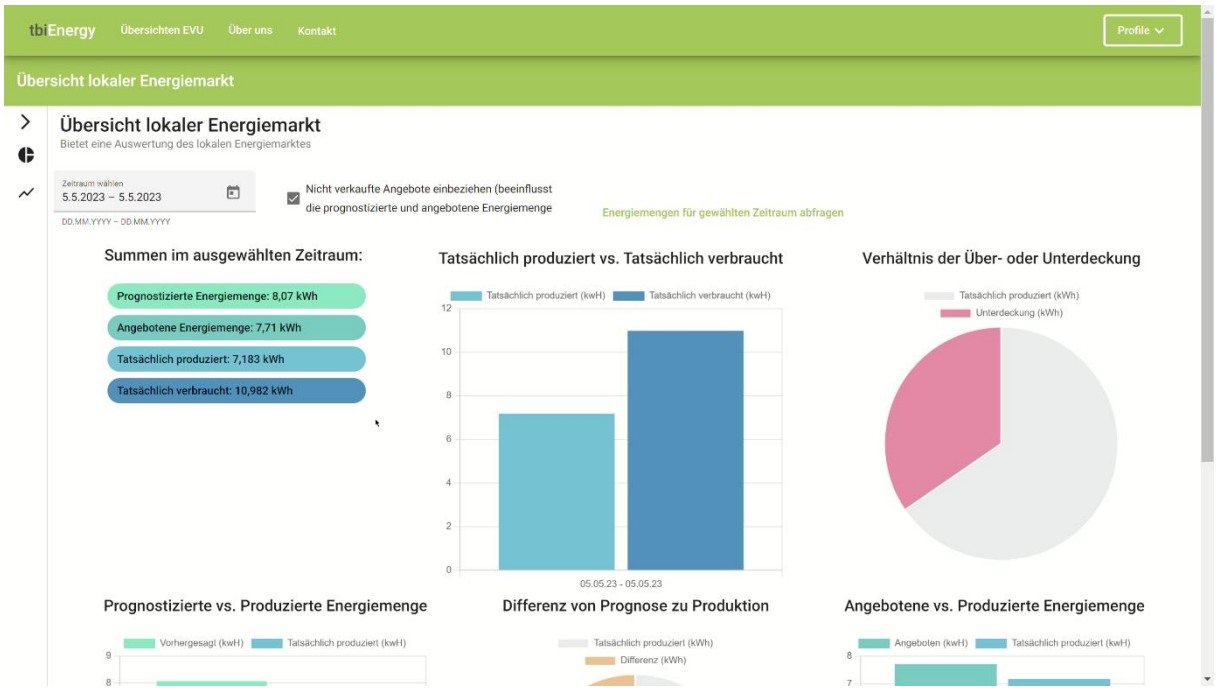

Abbildung 25: Bilanz anzeigen (Webportal)

7 Plattformintegration und Feldtest

Im Folgenden werden die Unterarbeitspakete vom AP 5 genauer beschrieben.

7.1 Plattformintegration

Im Rahmen des Arbeitspaketes wurden die Systemkomponenten der HSB in die in AWS-Infrastruktur des Projektpartners Arxum eingebunden. Die Einbindung erfolgte dabei unter der Verwendung von CI/CD-Pipelines, sodass neue Features und Anpassungen mit geringem Aufwand umgesetzt und veröffentlicht werden können. Abbildung 2626 gibt einen Überblick über die integrierten Systemkomponenten. Im Zuge der agilen Softwareentwicklung wurde das Gesamtsystem fortlaufend evaluiert und die notwendigen Anpassungen an den Teilkomponenten gemeinsam mit den Projektpartnern spezifiziert. Die Arbeiten wurden gemeinsam mit den Projektpartnern im Zuge des Feldtests abgeschlossen.

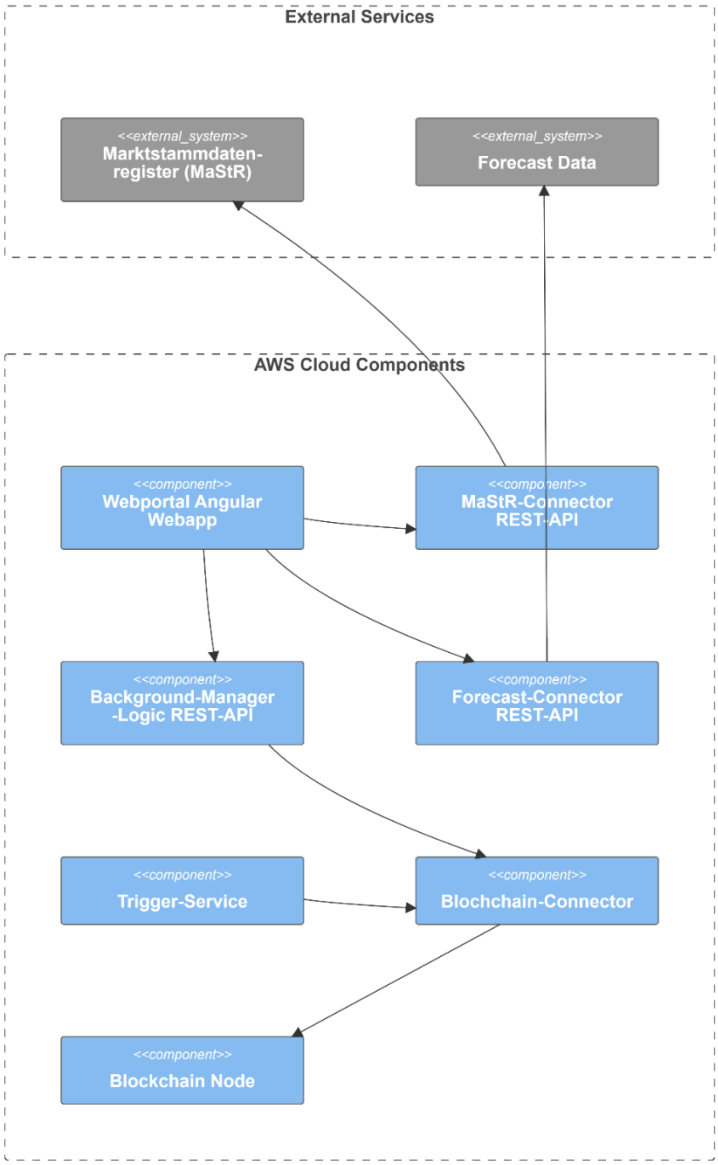

Abbildung 26: Übersicht integrierte Systemkomponenten

7.2 Erstellen von Testfällen

Im Rahmen des Arbeitspaketes wurde eine Liste von Testfällen anhand der in AP 2.1 enthaltenen Sollwerte erarbeitet und ein gemeinsamer Testkatalog (siehe Anhang) erstellt. Diese Testfälle wurden anschließend im Rahmen des Feldtests mit den Ist-Werten des entwickelten Demonstrators abgeglichen. Die Verarbeitungslogik und die Schnittstellen der durch die HSB entwickelten Microservices wurden fortlaufend im Zuge der Entwicklung mithilfe von Unit- und Integrations-Tests getestet.

7.3 Feldtest mit Modell Nutzern

Um das Ausbringen der Softwarekomponenten auf die Hardware der Feldtestteilnehmer zu unterstützen, wurden entsprechende Deployment-Skripte umgesetzt und den Projektpartnern zur Verfügung gestellt. Des Weiteren wurde im Webportal eine Übersicht über die angebundenen Hard- und Softwarekomponenten implementiert, um zum einen deren Erreichbarkeit zentral und schnell

überprüfen zu können und zum anderen um Fehler im Rahmen des Feldtests leichter beheben zu können.

Im Rahmen eines vorgelagerten Feldtests konnten alle Teilnehmer die Funktionen des Webportals für die ihnen zugewiesenen Rollen (User, Energieversorger, Administrator) erproben. Diese Vorerprobungen liefen erfolgreich ab. Zur abschließenden Überprüfung wurde während des eigentlichen Feldtests der bereits im Vorfeld definierte Testkatalog durchgespielt. Auch diese Prüfung verlief reibungslos und ohne besondere Vorkommnisse.

8 Nutzen und Verwertung

Die Verwertung ist in Form von Beratungsleistungen für z.B. Stadtwerke als Akteure in der Energiebranche angestrebt. Dabei stehen die Themen des lokalen Energiemarkts besonders im Fokus, und die gewonnenen Erkenntnisse aus den Themen IT-Sicherheit, Blockchain und Distributed-Ledger-Technologien werden angeboten. Ein besonders wichtiger Teil in der Verwertungskette einer praxisorientierten Forschungs- und Bildungseinrichtung ist der Beitrag zur Lehre durch die Aktualität der Themen aus den Forschungsprojekten. Auch spannende und aktuelle Themen für Abschlussarbeiten werden aus den Problemstellungen der Forschungsprojekte abgeleitet und liefern damit gleichzeitig einen Beitrag zur regionalen Kompetenzstärkung. Die Hochschule Bremen publiziert Ergebnisse aus den Forschungsprojekten auf nationalen und internationalen Konferenzen sowie in regionalen Foren.

9 Kommunikation von Fachinformationen

Im Rahmen des Forschungsprojekts wurden gewonnene Erkenntnisse in Form von Fachinformationen veröffentlicht und bereitgestellt.

9.1 Aktivitäten zur Veröffentlichung

Das durch die HSB eingereichte Abstract auf der 6. Blockchain Autumn School (BAS2022) zum Thema „A blockchain based local energy market" wurde akzeptiert und ein entsprechendes Full-Paper eingereicht, veröffentlicht und präsentiert. Die Veröffentlichung wurde dabei mit dem zweiten Platz des Best Paper Award ausgezeichnet.

9.2 Aktivitäten zur Standardisierung

Die HSB hat an der Standardisierung der Interoperabilität von DLT-Systemen in der ISO/TC 307 „Blockchain and distributed ledger technologies" in der Working Group 7 - Interoperability mitgewirkt.

Literaturverzeichnis

[1] BMWi/BSI, „Standardisierungsstrategie zur sektorübergreifenden Digitalisierung nach dem Gesetz zur Digitalisierung der Energiewende," 2019.

[2] G. Gritzan, T. Petrow, M. Jakobi, S. Knodel und R. Sethmann, „A blockchain-based local energy market," *Konferenzband zum Scientific Track der Blockchain Autumn School 2022 (BAS2022),* pp. 51-56, 2022.

[3] BSI. [Online]. Available: https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKompendium/itgrundschutzKompendium_node.html. [Zugriff am 25 09 2019].

[4] BSI. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Krypto/Blockchain_Analyse.pdf?__blob=publicationFile&v=5. [Zugriff am 25 09 2019].

[5] BSI. [Online]. Available: https://docs.microsoft.com/de-de/azure/security/develop/threat-modeling-tool-getting-started#reports--sharing. [Zugriff am 25 09 2019].

Anhang

Testkatalog tbiEnergy

Der folgende Testkatalog enthält eine Liste von Testfällen, deren Testschritte mit den enthaltenden Sollwerten aus den in AP 2.1 erarbeiteten Use Cases abgeleitet wurden. Diese Testfälle werden im Rahmen des Feldtests mit den Ist-Werten des entwickelten Demonstrators abgeglichen. Die Verarbeitungslogik und die Schnittstellen der verwendeten Microservices wurden bereits im Zuge der Entwicklung mit Hilfe von Unit- und Integration-Tests getestet.

01 Verwaltung CLS-Box, Units und Meter

Um die CLS-Boxen, Einheiten und Meter auf die spätere Anbindung an den blockchainbasierten lokalen Energiemarkt vorzubereiten, wurde eine entsprechende prototypische Verwaltungsstruktur umgesetzt, die mit den folgenden Testfällen überprüft wird.

01.1 Testfall - Add CLS Box

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC01.1-Req01	Cognito Admin Account ist vorhanden	Um eine CLS-Box hinzufügen zu können wird ein entsprechender Admin Account (Username, Passwort) im tbiEnergy Cognito Pool benötigt	☑ Erfüllt ☐ Nicht erfüllt

TC01.1-Req02	Die CLS-Box am Standort wurde durch einen Techniker installiert	Physische Installation vor Ort ist erfolgt und die IMEI der im Gerät befindlichen SIM-Karte wurde dokumentiert	☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req03	CLS-Box wurde installiert & bestromt		☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req04	Antenne angeschlossen		☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req05	SIM Karte im Gerät eingebaut	Nach dem Start des Gerätes sollte die LTE-Verbindung durch die am Gerät befindliche LTE-LED signalisiert werden (LTE LED ein, falls eine Verbindung besteht)	☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req06	Die CLS-Box-Logic wurde auf die CLS-Box deployed und ist erreichbar	Deployment der CLS-Box-Logic über einen Docker Remote Context ist erfolgt und die Erreichbarkeit validiert.	☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req07	Die Daten der CLS-Box wurden in Confluence erfasst	• IP-Adresse im IoT-VPN • Name des Gerätes • Aufstellungsort des Gerätes	☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req08	OpenEMS wurde deployed und konfiguriert.	Alle steuerbare Hardware vor Ort: PV-Wechselrichter, Zähler etc. wurde eingerichtet	☑ Erfüllt ☐ Nicht erfüllt
TC.01.1-Req09	OpenEMS WebUI ist unter http://lokaleip/ erreichbar (Historische Daten sind abrufbar)	Die WebUI lässt sich über den Browser aufrufen und etwaige Komponenten lassen sich noch integrieren.	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Hinzufügen der CLS-Box über das Webportal

Bezeichner	Testschritt	Beschreibung	Voraussetzung	Eingaben	Ausgaben	Ergebnis	Kommentar
TC01.1-TS01	Prüfung Betriebsfähigkeit CLS-Box	Prüfung der allgemeinen Betriebsfähigkeit über visuelle Kontrolle der Indikator-LEDs am Gerät.	TC01.1-Req02, TC01.1-Req03	Gerät bestromen	PWR LED leuchtet nach ca. 10s konstant (visuelle Kontrolle)	☑ Bestanden ☐ Nicht bestanden	
TC01.1-TS02	Prüfung der Kommunikationsfähigkeit der CLS-Box	Die Verfügbarkeit der LTE-Verbindung des Gerätes wird über die Indikator-LEDs am Gerät geprüft.	TC.01.1-Req04, TC.01.1-Req05	Gerät ist seit ca. 1 Minute in einem betriebsbereiten Zustand.	LTE LED leuchtet konstant	☑ Bestanden ☐ Nicht bestanden	
TC01.1-TS03	Prüfung der OpenEMS Komponente	Die Zustand der OpenEMS Software lässt sich über die WebUI testen.	TC.01.1-Req08	Lokale IP Adresse der CLS-Box (Gerät hat über einen DHCP-Server eine Adresse über den WAN-Port bezogen oder ist unter der statischen Adresse 192.168.137.110 erreichbar)	Steuerbare Geräte werden live und über historische Werte in der WEB-Ui angezeigt.	☑ Bestanden ☐ Nicht bestanden	
TC01.1-TS04	Login im Webportals als Administrator	Ein Administrator loggt sich im Webportal ein und ruft das Entwickler Dashboard auf	TC.01.1-Req01	• Admin Cognito Account	Entwickler-Dashboard wird angezeigt	☑ Bestanden ☐ Nicht bestanden	
TC01.1-TS05	Durchführung des Dialogs zum Hinzufügen der CLS-Box	Ein Administrator ist im Webportal eingeloggt, ruft das Entwicklerdashboard (Background-Manager-Logic) auf und startet den Dialog “CLS-Box hinzufügen”. Unter Angabe der in Confluence erfassten Daten wird die Erreichbarkeit der CLS-Box geprüft und bei erfolgreicher Verbindung ein entsprechender Datenbankeintrag angelegt.	TC.01.1-Req06, TC.01.1-Req07	• Admin Cognito Account • IP-Adresse im IoT-VPN • Name des Gerätes • Aufstellungsort des Gerätes	Dialog zeigt an, dass die CLS-Box erfolgreich hinzugefügt wurde	☑ Bestanden ☐ Nicht bestanden	

TC01.1-TS06	Überprüfung ob die CLS-Box erfolgreich zum Verwaltungssystem hinzugefügt wurde	Ein Administrator ist im Webportal eingeloggt, ruft das Entwickler Dashboard (Background-Manager-Logic) auf und startet den Dialog “Liste aller verbundenen CLS-Boxen”.	TC01.1-TS03	-	CLS-Box sollte in der Liste enthalten sein	☑ Bestanden ☐ Nicht bestanden	
TC01.1-TS07	Kontrolle der Erreichbarkeit der CLS-Box in der Übersicht der angebundenen	Ein Administrator ist im Webportal eingeloggt, ruft das Entwickler Dashboard (Komponentenübersicht) auf und überprüft, ob die hinzugefügte CLS-Box vorhanden ist.	TC.01.1-Req*	• Admin Cognito Account	Der Health Status der CLS-Box ist online	☑ Bestanden ☐ Nicht bestanden	

01.2 Testfall - Add Unit

Vorrausetzungen

Bezeichner	Voraussetzung	Beschreibung	Ergebnis
TC01.2-Req01	01.1 Testfall - Add CLS Box bestanden	Die Vorrausetzungen sind erfüllt und alle Testschritte wurden bestanden	☑ Erfüllt ☐ nicht erfüllt
TC01.2-Req02	Die MaStR-Nr. und die MaStR-Nr. des Anlagenbetreibers der Einheit wurde erfasst	Die Daten des Teilnehmers wurden in Confluence erfasst	☑ Erfüllt ☐ Nicht erfüllt
TC01.2-Req03	OpenEMS erreichbar	OpenEMS wurde auf der CLS-Box gestartet und ist erreichbar	☑ Erfüllt ☐ Nicht erfüllt
TC01.2-Req04	Anbindung an OpenEMS	Anbindung der Einheit an OpenEMS ist erfolgt	☑ Erfüllt ☐ Nicht erfüllt
TC01.2-Req05	Steuerbare Einheiten sind in OpenEMS gemäß ihrer Nomenklatur im MaStR benannt.	Als Voraussetzung für die CLS-Box Logic müssen die Einheiten über einen REST-Call auf OpenEMS erreichbar sein. Dies bedingt, dass bspw. Wechselrichter gemäß ihrer Markstammdatennummer hinterlegt sind.	☑ Erfüllt ☐ Nicht erfüllt
TC01.2-Req06	Administrationsrechte im Webportal	Admin Account (Username, Passwort) im tbiEnergy Cognito Pool	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Hinzufügen einer Einheit über das Webportal

Bezeichner	Testschritt	Beschreibung	Voraussetzung	Eingaben	Ausgaben	Ergebnis	Kommentar
TC01.2-TS01	Aufruf Dialog	Aufruf “Entwickler Dashboard” > “Background-Manager-Logic” > “Einheit hinzufügen”	TC01.2-Req06	-	Dialog “Einheit hinzufügen” öffnet sich	☑ Bestanden ☐ Nicht bestanden	

TC01.2-TS02	Auswahl CLS-Box	Auswahl einer CLS-Box deren Einheit hinzugefügt werden soll	TC01.2-Req01	• Auswahl CLS-Box anhand von
				  ◦ IP-Adresse im IoT-VPN
				  ◦ Name des Geräts
				  ◦ Aufstellungsort des Geräts,
				• dann “CLS-Box auswählen” klicken
					Anzeige des Tabs “Angabe der Marktakteursnummer”	☑ Bestanden
						☐ Nicht bestanden	

TC01.2-TS03	Angabe Marktakteursnummer	Erfassung der MaStR-Nr. des Anlagenbetreibers	TC01.2-Req02, TC01.2-Req03, TC01.2-Req04, TC01.2-Req05	Eingabe der MaStR-Nr. des Anlagenbetreibers, dann “Marktakteursnummer übergeben” klicken	Anzeige des Tabs “Auswahl der Einheit”	☑ Bestanden ☐ Nicht bestanden	

TC01.2-TS04	Auswahl der Einheit	Auswahl der Einheit aus der Liste von Einheiten des spezifizierten Anlagenbetreibers.	TC01.2-Req02, TC01.2-Req03, TC01.2-Req04, TC01.2-Req05	Auswahl anhand der MaStR-Nr. der Einheit, dann “Einheit auswählen” klicken	Anzeige des Tabs “Einheit hinzufügen”	☑ Bestanden ☐ Nicht bestanden	
TC01.2-TS05	Erreichbarkeit	Abschließend erfolgt die Prüfung ob die angegebene Einheit an der ausgewählten CLS-Box erreicht werden kann.	TC01.2-Req02, TC01.2-Req03, TC01.2-Req04, TC01.2-Req05	Einheit hinzufügen	Meldung, dass die Einheit erfolgreich hinzugefügt wurde	☑ Bestanden ☐ Nicht bestanden	
TC01.2-TS06	Prüfung	Aufruf “Entwickler Dashboard” > “Background-Manager-Logic” > “Liste aller hinzugefügten Einheiten”	TC01.2-TS01, TC01.2-Req06	-	Einheit ist aufgeführt	☑ Bestanden ☐ Nicht	

01.3 Testfall - Add Meter

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC01.3-Req01	Meter angebunden	Anbindung des Meters an OpenEMS ist erfolgt	☑ Erfüllt ☐ Nicht erfüllt
TC01.3-Req02	Administrationsrechte im Webportal	Admin Account (Username, Passwort) im tbiEnergy Cognito Pool	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Hinzufügen eines Meters über das Webportal							
Bezeichner	**Testschritt**	**Beschreibung**	**Voraussetzung**	**Eingaben**	**Ausgaben**	**Ergebnis**	**Kommentar**
TC01.3-TS01	Aufruf Dialog	Aufruf “Entwickler Dashboard” > “Background-Manager-Logic” > “Meter hinzufügen”	TC01.3-Req02	-	Dialog “Meter hinzufügen” öffnet sich	☑ Bestanden ☐ Nicht bestanden	

TC01.3-TS02	Auswahl CLS-Box	Auswahl einer CLS-Box deren Meter hinzugefügt werden soll	TC01.1-TS03, TC01.2-Req03, TC-01.03-Req01	• Auswahl der CLS-Box anhand von
				  ◦ IP-Adresse im IoT-VPN
				  ◦ Name des Geräts
				  ◦ Aufstellungsort des Geräts
				• “CLS-Box auswählen” klicken
					Tab “Angabe der Seriennummer des Meters” öffnet sich	☑ Bestanden
						☐ Nicht bestanden	

TC01.3-TS03	Eingabe Seriennummer	Angabe der Seriennummer des Meters.	TC01.1-TS03, TC01.2-Req03, TC-01.03-Req01	SN-Meter, dann “Seriennummer übergeben” klicken	Tab “Meter hinzufügen” öffnet sich	☑ Bestanden ☐ Nicht bestanden	
TC01.3-TS04	Meter hinzufügen	Eine Zusammenfassung wird gezeigt die bestätigt werden muss.	-	Prüfen und “Meter hinzufügen” klicken	Meldung, dass das Meter erfolgreich hinzugefügt wurde	☑ Bestanden ☐ Nicht bestanden	
TC01.3-TS05	Prüfung	Aufruf “Entwickler Dashboard” > “Background-Manager-Logic” > “Liste aller verbundenen Meter”	TC01.3-Req02	-	Meter ist in der Liste	☑ Bestanden ☐ Nicht bestanden	

01.4 Testfall - Register EOSIO User

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC01.4-Req01	TPM_API wurde auf der CLS-Box gestartet und ist erreichbar		☑ Erfüllt ☐ Nicht erfüllt
TC01.4-Req02	Administrationsrechte im Webportal	Admin Account (Username, Passwort) im tbiEnergy	☑ Erfüllt

Testfall - Registrierung einer Einheit über das Webportal							
Bezeichner	**Testschritt**	**Beschreibung**	**Voraussetzung**	**Eingaben**	**Ausgaben**	**Ergebnis**	**Kommentar**
TC01.4-TS01	Öffnen des Dialogs	Aufruf “Entwickler Dashboard” > “Background-Manager-Logic” > “Registrierung eines Eos Accounts”	TC01.1-TS03, TC01.4-Req*	-	Dialog “Registrierung eines neuen EOS Accounts durch einen Administrator” öffnet sich	☑ Bestanden ☐ Nicht bestanden	
TC01.4-TS02	Bestätigung des Accounts	Auswahl der CLS-Box aus der Liste der bereits hinzugefügten CLS-Boxen und Angabe des EOS Account-Namens	TC01.1-TS03, TC01.4-Req*	• CLS-Box-IP • Name des EOS Accounts • Auswahl Button “Registrieren”	Meldung, dass der Account erfolgreich registriert wurde	☑ Bestanden ☐ Nicht bestanden	

02 Vorgänge lokaler Energiemarkt

Nachdem die Anbindung an den blockchainbasierten, lokalen Energiemarkt erfolgreich durchgeführt wurde, können die Einheiten auf dem lokalen Energiemarkt registriert, Angebote platziert und Gebote abgegeben werden. Außerdem kann dem Nutzer eine Übersicht über die tatsächliche produzierte Energiemenge gegeben und der erfolgte Verbrauch damit abgeglichen werden. Die folgenden Testfälle decken zum einen die Marktprozesse und zum anderen den Ablauf der Übertragung der Erzeugungs- und Verbrauchsdaten ab. Es wird vorausgesetzt, dass alle Testfälle aus Abschnitt 01 erfolgreich durchgeführt und bestanden wurden.

02.1 Testfall - Register Unit on local market

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC02.1-Req01	Testfälle im Testabschnitt “01 01 Verwaltung CLS-Box, Units und Meter” alle bestanden	Nur wenn alle Testfälle erfolgreich durchgeführt wurden kann die Einheit auf dem lokalen Energiemarkt registriert werden	☑ Erfüllt ☐ Nicht erfüllt
TC02.1-Req02	Administrationsrechte im Webportal	Admin Account (Username, Passwort) im tbiEnergy Cognito Pool	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Registrierung einer Einheit auf dem lokalen Energiemarkt							
Bezeichner	**Testschritt**	**Beschreibung**	**Voraussetzung**	**Eingaben**	**Ausgaben**	**Ergebnis**	**Kommentar**
TC02.1-TS01	Aufruf des Dialogs	Aufruf “Entwickler Dashboard” > “Background-Manager-Logic” > “Registrierung einer Einheit auf dem lokalen Energiemarkt”	TC01.1-TS03, TC02.1-Req*	-	Anzeige des Dialogs “Registrierung einer Einheit auf dem lokalen Energiemarkt”	☑ Bestanden ☐ Nicht bestanden	
TC02.1-TS02	Auswahl Einheit	Auswahl einer Einheit aus der Liste der bereits hinzugefügten Einheiten anhand ihrer MaStR-Nr	TC01.1-TS03, TC02.1-Req*	• MaStR-Nr. der Einheit • Klick auf “Einheit registrieren”	Meldung, dass die Einheit erfolgreich registriert wurde	☑ Bestanden ☐ Nicht bestanden	

02.2 Testfall - Place Offer on local market

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC02.2-Req01	Cognito User Account ist vorhanden	User Account (Username, Passwort) im tbiEnergy Cognito Pool benötigt mit den folgenden Attributen: • Betreiber-MaStR-Nr • Eos-Account • CLS-Box-IP • kwhAnnum	☑ Erfüllt ☐ Nicht erfüllt
TC02.2-Req02	TC02.1-TS01 bestanden	Für den Nutzer wurde durch einen Administrator eine Einheit auf dem lokalen Energiemarkt registriert	

Testfall - Angebot auf dem lokalen Energiemarkt platzieren							
Bezeichner	**Testschritt**	**Beschreibung**	**Voraussetzung**	**Eingaben**	**Ausgaben**	**Ergebnis**	**Kommentar**
TC02.2-TS01	Login im Webportal als User	Ein User loggt sich im Webportal ein	TC.02.2-Req01	User Cognito Account	Lokaler Energiemarkt wird angezeigt	☑ Bestanden ☐ Nicht bestanden	
TC02.2-TS02	Einstellen eines Angebotes starten	Aufruf “Lokaler Energiemarkt” > “Angebot einstellen”	TC.02.2-Req*	-	Dialog “Angebot platzieren” öffnet sich und zeigt die auf dem Energiemarkt registrierten Einheiten	☑ Bestanden ☐ Nicht bestanden	
TC02.2-TS03	Auswahl Einheit	Auswahl der Einheit für die Energie angeboten werden soll	-	MaStR-Nr. oder Name der Einheit	Wechsel zum Schritt “Angebotszeitpunkt”	☑ Bestanden ☐ Nicht bestanden	
TC02.2-TS04	Auswahl Angebotszeitpunkt	Auswahl des Start-Zeitpunktes (Datum und Uhrzeit) ab dem für einen Slot von 15 Minuten Energie angeboten werden soll	-	Eingabe eines Datums und einer Uhrzeit (Angebotszeitpunkt)	Wechsel zum Schritt “Startpreis und Energiemenge”	☑ Bestanden ☐ Nicht bestanden	
TC02.2-TS05	Auswahl Preis und Energiemenge	Angabe der Energiemenge die angeboten werden soll. Die Menge ist beschränkt auf das vom Forecast-Connector berechnete Maximum. Angabe eines gewünschten Startpreises (Minimalpreis).	-	Auswahl der Energiemenge und Angabe des Minimalpreises (Startpreis)	Wechsel zum Schritt “Bestätigung des Angebotes”	☑ Bestanden ☐ Nicht bestanden	
TC02.2-TS06	Bestätigung	Zusammenfassung des Angebotes	-	Bestätigung der Angaben mit “Angebot einstellen“	Meldung, dass das Angebot erfolgreich platziert wurde	☑ Bestanden ☐ Nicht bestanden	
TC02.2-TS07	Prüfung	Auswahl “Lokaler Energiemarkt” > Übersicht der Angebote	-	-	Die Ansicht enthält das eingestellte Angebot	☑ Bestanden ☐ Nicht bestanden	

02.3 Testfall - Place Bid on local market

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC02.3-Req01	Cognito User Account ist vorhanden	User Account bestehend aus Benutzername und Passwort	☑ Erfüllt ☐ Nicht erfüllt

TC02.3-Req02	User ist eingeloggt	Die Anmeldung am Webportal ist erfolgt	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Bieten auf Angebote							
Bezeichner	**Testschritt**	**Beschreibung**	**Voraussetzung**	**Eingaben**	**Ausgaben**	**Ergebnis**	**Kommentar**
TC02.3-TS01	Anzeige der kaufbaren Angebote	Auswahl “Lokaler Energiemarkt” > “Gebot erstellen”	TC02.3-Req*	-	Anzeige der verfügbaren Angebote	☑ Bestanden ☐ Nicht bestanden	
TC02.3-TS02	Auswahl eines Angebotes und Gebot	Auswahl eines erwerbbaren Angebotes und Eingabe eines Gebotes.	-	Eingabe eines Preises über dem aktuellen Höchstgebot	Erfolgreiche Platzierung des Gebotes	☑ Bestanden ☐ Nicht bestanden	
TC02.3-TS03	Prüfung	Auswahl “Lokaler Energiemarkt” > Übersicht der Gebote	-	-	Die Ansicht enthält das abgegebene Gebot	☑ Bestanden ☐ Nicht bestanden	

02.4 Testfall - Trigger and Execution Event (send_data)

Vorrausetzungen			
Bezeichner	**Voraussetzung**	**Beschreibung**	**Ergebnis**
TC02.4-Req01	EOS tbiEnergy Admin Account	Ein EOS Account der dazu berechtigt ist die im Rahmen des Projekts entwickelten Smart Contracts aufzurufen	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Auslösen und Verarbeitung des Events zur Übermittlung der Erzeugungs- und Verbrauchsdaten							
Bezeichner	**Testschritt**	**Beschreibung**	**Voraussetzung**	**Eingaben**	**Ausgaben**	**Ergebnis**	**Kommentar**
TC02.4-TS01	Login	Autorisierung mit Hilfe der “login” Route (auth/login) der tbiEnergy Thunder Collection	TC02.4-Req01	• E-Mail • Passwort	Gültiges Auth-Token	☑ Bestanden ☐ Nicht bestanden	
TC02.4-TS02	Manuelles auslösen des Events	Mit Hilfe der tbiEnergy Thunder Collection wird die “trigger event” Route (performer/trigger event) wird ein Event auf der Blockchain ausgelöst	TC02.4-TS01, TC02.4-Req02	• Auth-Token • Blockchain Accountname (tbiAdmin) • Event Typ • Einheit oder Seriennummer des Meters • Zeit-Slot • Verbrauchs oder Erzeugungswert • EOS Name des Käufers	Response mit Http-Statuscode 200	☑ Bestanden ☐ Nicht bestanden	

TC02.4-TS03	Verarbeitung des Events in den	Der Watcher auf der CLS-Box leitet das Event an die CLS-Box-Logic	TC02.4-TS01, TC02.4-Req02	• Blockchain Accountname	Im Webportal kann	☑ Bestanden	

02.5 Testfall - Balancing Group Overview

Vorrausetzungen

Bezeichner	Voraussetzung	Beschreibung	Ergebnis
TC02.5-Req01	Cognito EVU Account ist vorhanden	EVU Account (Username, Passwort) im tbiEnergy Cognito Pool.	☑ Erfüllt ☐ Nicht erfüllt

Testfall - Bilanzkreisübersicht lokaler Energiemarkt

Bezeichner	Testschritt	Beschreibung	Voraussetzung	Eingaben	Ausgaben	Ergebnis	Kommentar
TC02.5-TS01	Login im Webportal als EVU	Ein EVU loggt sich im Webportal ein	TC.02.2-Req01	User Cognito Account in der Gruppe EVU	Die Übersicht übern den Bilanzkreis des lokalen Energiemarkts wird angezeigt	☑ Bestanden ☐ Nicht bestanden	
TC02.5-TS02	Kontrolle des Bilanzkreise des lokalen Energiemarkts	Nachdem das EVU sich eingeloggt hat wird direkt eine Übersicht über die produzierten und verbrauchten Energiemengen der am lokalen Energiemarkt partizipierenden Akteure angezeigt.	TC02.5-TS01	-	Die Übersicht übern den Bilanzkreis des lokalen Energiemarkts wird angezeigt	☑ Bestanden ☐ Nicht bestanden	

Berichtsblatt

1. ISBN oder ISSN geplant	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht

3. Titel
Abschlussbericht zum Verbundvorhaben: Trusted Blockchains für das offene, intelligente Energienetz der Zukunft (tbiEnergy)
Teilvorhaben: IT-Sicherheit für Trusted Blockchains, im intelligenten Energienetz der Zukunft (ITSitbiE)

4. Autor(en) [Name(n), Vorname(n)]	5. Abschlussdatum des Vorhabens
Sethmann, Richard; Gritzan, Giacomo; Jakobi, Michelle; Petrow, Torben; Knodel, Sibille; Albers, Julia	31.05.2023
	6. Veröffentlichungsdatum
	7. Form der Publikation Document Control Sheet
8. Durchführende Institution(en) (Name, Adresse) Hochschule Bremen - Fachbereich Elektrotechnik und Informatik - Institut für Informatik und Automation	**9. Ber.-Nr. Durchführende Institution**
	10. Förderkennzeichen 03EI6029B
	11. Seitenzahl 52
12. Fördernde Institution (Name, Adresse) BMWK	**13. Literaturangaben** 5
	14. Tabellen 2
	15. Abbildungen 26

16. DOI (Digital Object Identifier)

17. Vorgelegt bei (Titel, Ort, Datum)

18. Kurzfassung
Das Projekt tbiEnergy setzt auf dem bestehenden regulierten Energiemarkt auf und adressiert mit einem ganzheitlichen Blockchain-Ansatz die momentan in den Energienetzen anstehende Problematik der effektiven Integration alternativer Energieerzeugung in bestehende Netzstrukturen. Außerdem soll die aktuelle Lücke der fehlenden Komfortdienste des heutigen intelligenten Stromnetzes durch den Einsatz der Blockchain geschlossen werden. Mit Hilfe der Blockchain-Technologie und den in ihr formulierten Smart-Contracts, lassen sich innovative Geschäftsmodelle auch ohne hohe Investitionen in die IKT oder Softwareinfrastruktur bei gleichzeitiger inhärenter Sicherheit realisieren. Am Markt befindliche Blockchain-Lösungen sind allerdings bisher nicht explizit für den energiewirtschaftlichen Einsatz konzipiert. Es fehlen Kundenschnittstellen, eine Anbindung an die Infrastruktur der Energieversorger sowie ein stringentes Hardwaresicherheitskonzept. Über das Mehrwertdienstekonzept des deutschen Smart-Meter-Gateways lassen sich die Vorstellungen des Bundesamt für Sicherheit in der Informationstechnik (BSI) und die Vorzüge einer kryptographisch gesicherten, verteilten Datenbank in tbiEnergy zusammenführen. Dazu finden sich im Konsortium Hardwaresicherheitsexperten wie die Infineon AG, die devolo AG als Hersteller von Smart-Grid-Hardware und -Lösungen, die Hochschule Bremen als ausgewiesene Expertin im Thema IT-Sicherheit, dem Blockchain-Startup Arxum GmbH und die Stadtwerkte Trier AöR als Anwendungspartner zusammen. Es wurde eine Plattform geschaffen, die generische Geschäftsprozesse innerhalb einer Blockchain abbildbar macht und sich dennoch in die aktuellen energiewirtschaftlichen Regularien einfügt.
Die im Rahmen des Projektes erarbeiteten Konzepte kulminieren in einem abschließenden Demonstrator, der im Rahmen eines Feldtests anhand eines Testkatalogs überprüft wird.

19. Schlagwörter
DLT, Blockchain, lokaler Energiemarkt, Energiewende

20. Verlag	21. Preis
Technische Informationsbibliothek (TIB)	

Nicht änderbare Endfassung mit der Kennung 1918111-8

Document control sheet

1. ISBN or ISSN planned	**2. type of document (e.g. report, publication)** Veröffentlichung (Publikation)
3. title Final Report on the joint project: Trusted Blockchains for the Open, Intelligent Energy Grid of the Future (tbiEnergy) <b/> Subproject: IT Security for Trusted Blockchains, in the Intelligent Energy Grid of the Future (ITSitbiE)	
4. author(s) (family name, first name(s)) Sethmann, Richard; Gritzan, Giacomo; Jakobi, Michelle; Petrow, Torben; Knodel, Sibille; Albers, Julia	**5. end of project** 31.05.2023
	6. publication date
	7. form of publication Document Control Sheet
8. performing organization(s) name, address Hochschule Bremen - Fachbereich Elektrotechnik und Informatik - Institut für Informatik und Automation	**9. orginators report no.**
	10. reference no. 03EI6029B
	11. no. of pages 52
12. sponsering agency (name, address) BMWK	**13. no. of references** 5
	14. no. of tables 2
	15. no. of figures 26
16. DOI (Digital Object Identifier)	
17. presented at (title, place, date)	
18. abstract The tbiEnergy project is based on the existing regulated energy market and uses a holistic blockchain approach to address the current problem of effectively integrating alternative energy generation into existing grid structures. In addition, the current gap in the lack of convenience services in today's smart grid is to be closed through the use of blockchain. With the help of blockchain technology and the smart contracts formulated in it, innovative business models can be realized without high investments in ICT or software infrastructure while maintaining inherent security. However, blockchain solutions on the market have not yet been explicitly designed for the use in the energy industry. There is a lack of customer interfaces, a connection to the infrastructure of energy suppliers and a stringent hardware security concept. The value-added service concept of the German smart meter gateway (SMGW) allows the ideas of the German Federal Office for Information Security (BSI) and the advantages of a cryptographically secured, distributed database to be combined in tbiEnergy. The consortium brings together hardware security experts such as Infineon AG, devolo AG as a manufacturer of smart grid hardware and solutions, Bremen University of Applied Sciences as a proven expert in IT security, the blockchain startup Arxum GmbH and Stadtwerkte Trier AöR as an application partner. A platform was created that enables generic business processes to be mapped within a blockchain and yet still fits in with current energy industry regulations. The concepts developed as part of the project will culminate in a final demonstrator, which will be tested as part of a field test using a test catalog.	
19. keywords DLT, blockchain, local energy market, energy transition	
20. publisher German National Library of Science and Technology (TIB)	**21. price**

Nicht änderbare Endfassung mit der Kennung 1918118-5