

INHALT

1	Ziele	2
1.1	Hauptziel.....	2
1.2	Unterziele	2
2	Scope	2
2.1	Stakeholders	2
2.2	Systemkontext.....	3
3	Anforderungen.....	4
3.1	Übersicht	4
3.2	Funktionale Anforderungsübersicht.....	4
3.2.1	Detailbeschreibung.....	5
3.3	Nicht-Funktionale Anforderungsübersicht.....	6
3.3.1	Detailbeschreibungen.....	6
4	Glossar.....	7

1 ZIELE

Wir haben folgendes Hauptziel für das Projekt identifiziert:

1.1 HAUPTZIEL

HZ. Am Ende des Projekts 1 besteht eine Lösung, welche es Personen erlaubt Analysen der Blockchain zu realisieren, um diese zu optimieren.

1.2 UNTERZIELE

UZ-1. Die Source Codes sind von Projektbeginn an öffentlich im Internet zugänglich.

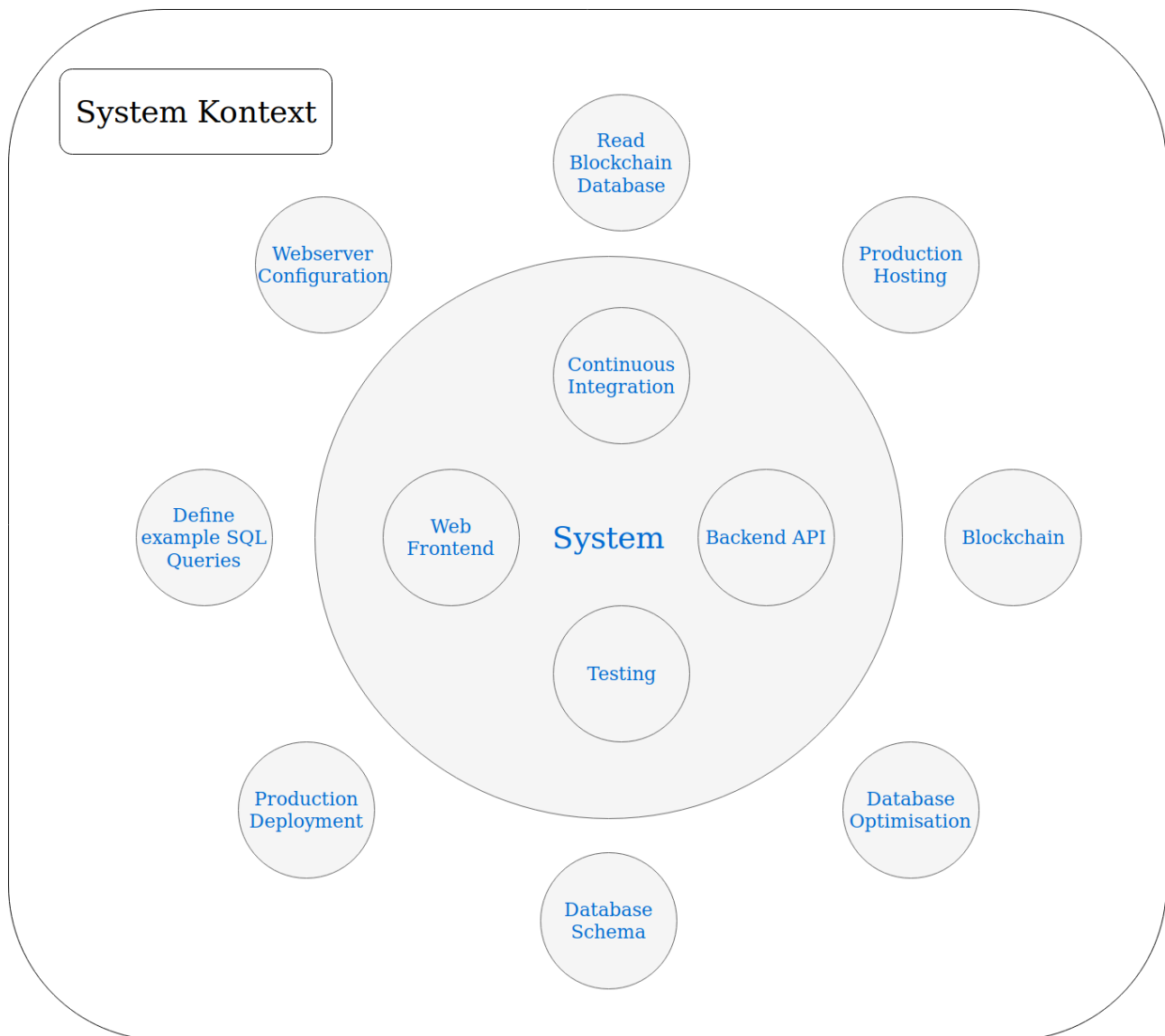
UZ-2. Die Lösung ist zu Projektende performant.

2 SCOPE

2.1 STAKEHOLDERS

- Kai Brännler (Kunde)
- Devs / Projektleitung
 - Elias Summermatter
 - Jan Friedli
- Anwender*innen
 - Studierende im Blockchain Modul
 - Forschende

2.2 SYSTEMKONTEXT



Das Systemkontext Diagramm dient der Modellierung der Systemumgebung und bietet in einer frühen Projektphase einen Überblick über das Projekt und die angrenzenden Systeme. Es soll aufzeigen, welche Abhängigkeiten ein Projekt besitzt und so schon frühzeitig potenzielle Problemstellen bei denen Abhängigkeiten sichtbar machen.

3 ANFORDERUNGEN

3.1 ÜBERSICHT

1 Funktionale Anforderungen

FA.1.1 Abfrage ausführen

FA .1.2 Keywords

FA.1.3 Suchfunktion

FA.1.4 Import/Export

FA.1.5 Details

FA.1.6 Elemente

2 Nicht-Funktionale Anforderungen

NFA.2.1 Open Source

NFA.2.2 Reaktionszeit

3.2 FUNKTIONALE ANFORDERUNGSÜBERSICHT

Nr.	Ziel	Kurzbezeichnung	Priorität	Komplexität	Risikobeurteilung
FA. 1.1	HZ	Wenn ein User eine Abfrage an das System stellt, muss das System fähig sein, diese selbständig auszuführen.	3	3	3
FA. 1.2	HZ	Wenn das System einen Userinput erhält, muss es bekannte Keywords hervorheben.	1	1	1
FA. 1.3	HZ	Das System muss die Möglichkeit bieten nach einzelnen Elementen zu suchen.	2	2	2
FA. 1.4	HZ	Das System sollte den Userinput exportieren/importieren können.	2	1	1.5
FA. 1.5	HZ	Das System kann alle Attribute eines Elements darstellen.	3	2	2.5
FA. 1.6	HZ	Das System wird die Möglichkeit bieten alle Elemente einzusehen.	3	3	3

Risikobeurteilungs-Formel: (Priorität + Komplexität) / 2 = Risikobeurteilung

Priorität $\in \{1,2,3\}$ wobei 3 die Höchste ist. Danach absteigend.

Komplexität $\in \{1,2,3\}$ wobei 3 die Höchste ist. Danach absteigend.

3.2.1 Detailbeschreibung

FA.1.1 Abfrage ausführen

Es wird eine Schnittstelle zu einem zweiten System bereitgestellt. Der User kann damit interagieren, indem er anhand der Schnittstelle Abfragen macht. Dieses zweite System verarbeitet die Anfragen und das System reicht die Anfragen an den User weiter. Das System erkennt Fehler und informiert den User entsprechend.

NF.1.2 Keywords

User können Eingaben machen, welche vom System so aufbereitet werden, dass diese gut lesbar sind. Dabei verarbeitet es ein vordefiniertes Set an Keywords. Bei jeder Änderung der Eingabe, wird diese neu evaluiert.

NF.1.3 Suchfunktion

Das System erlaubt es dem User nach einem Element zu suchen. Dabei kann nach verschiedenen Werten der diversen Attribute gesucht werden.

Falls es keine Treffer gab wird dies dem User mitgeteilt.

NF.1.4 Import/Export

Da User gegebenenfalls Abfragen mehrmals ausführen wollen, müssen diese exportiert werden können. Umgekehrt muss es auch möglich sein exportierte Abfragen wieder zu importieren.

NF.1.5 Details

Alle Attribute eines Elements können durch den User eingesehen werden. Falls das gewünschte Element nicht existiert, wird dies dem User mitgeteilt.

NF.1.6 Elemente

Das System erlaubt es dem User sich alle Elemente anzusehen.

3.3 NICHT-FUNKTIONALE ANFORDERUNGSÜBERSICHT

Nr.	Ziel	Kurzbezeichnung	Priorität	Komplexität	Risikobeurteilung
NFA.2.1	UZ-1	Das System muss unter einer Open Source Lizenz veröffentlicht werden.	3	1	2
NFA.2.2	UZ-2	Das System muss eine Reaktionszeit von unter 300ms bei Anfragen der maximalen Grösse von 10'000 Zeilen bieten.	2	3	2.5

Risikobeurteilungs-Formel: (Priorität + Komplexität) / 2 = Risikobeurteilung

Priorität $\in \{1,2,3\}$ wobei 3 die Höchste ist. Danach absteigend.

Komplexität $\in \{1,2,3\}$ wobei 3 die Höchste ist. Danach absteigend.

3.3.1 Detailbeschreibungen

NFA.2.1 Open Source

Alle entwickelten Software Komponenten werden auf GitHub veröffentlicht. Diese werde mittels AGPL lizenziert.

NFA.2.2 Reaktionszeit

Das System reagiert unter 300ms bei Anfragen der Grösse $\leq 10'000$ Zeilen. Anfragen die diese Grösse überschreiten, dürfen länger dauern.

4 GLOSSAR

API	Application Programming Interface Eine Schnittstelle für Programmierer*innen
GitHub	Plattform für Source Code Verwaltung
AGPL	GNU Affero General Public License
Open Source	Öffentlich zugänglicher Source Code
Blockchain	Eine wachsende Liste von sogenannten Blöcken, welche kryptographisch voneinander abhängig sind.