|  |
| --- |
|  |
|  |

|  |  |
| --- | --- |
| THEIA Baustellen App | |
| Eine Lösungsarchitektur für eine mobile Applikation | |
| **Masterthesis** | |
| Studiengang: | MAS-IT Software Architecture |
| Autor\*in: | Zürcher Amos | Software Architect |
| Betreuer\*in: | Staub Bill | Business Unit Lead “Business Solutions” |
| Auftraggeber\*in: | Hellion AG |
| Expert\*in: | Kleiner Helmut Max, Schmidhauser Arno |
| Datum: | [Datum einfügen] |

Management Summary

**Ausgangslage**

**Ziel**

**Vorgehen**

**Ergebnisse**

Inhaltsverzeichnis

[Eine Lösungsarchitektur für eine mobile Applikation 1](#_Toc188879106)

[Management Summary 2](#_Toc188879107)

[Inhaltsverzeichnis 3](#_Toc188879108)

[1 Einleitung 4](#_Toc188879109)

[1.1 Ausganslage und Motivation 4](#_Toc188879110)

[1.2 Problemstellung 4](#_Toc188879111)

[1.3 Ziel der Arbeit 4](#_Toc188879112)

[1.4 Abgrenzung 5](#_Toc188879113)

[1.5 Aufbau der Arbeit 5](#_Toc188879114)

[2 Literaturreview 5](#_Toc188879115)

[2.1 Grundlagen 5](#_Toc188879116)

[2.2 CAP Theorem 6](#_Toc188879117)

[2.2.1 Konsistenz 6](#_Toc188879118)

[2.2.2 Verfügbarkeit 6](#_Toc188879119)

[2.2.3 Partitionstoleranz 6](#_Toc188879120)

[2.3 Entwurfsmuster für die Datensynchronisation 7](#_Toc188879121)

[2.3.1 Datensynchronisation 7](#_Toc188879122)

[2.3.2 Datenspeicher und Verfügbarkeit 8](#_Toc188879123)

[2.3.3 Datenübertragung 8](#_Toc188879124)

[3 Methodik 8](#_Toc188879125)

[3.1 Überschrift 9](#_Toc188879126)

[3.1.1 Überschrift 9](#_Toc188879127)

[3.2 Überschrift 9](#_Toc188879128)

[4 Ergebnisse 9](#_Toc188879129)

[4.1 Pflichtenheft 9](#_Toc188879130)

[4.1.1 Authentifizierung 9](#_Toc188879131)

[4.1.2 Projektübersicht 10](#_Toc188879132)

[4.1.3 Projektdetails 10](#_Toc188879133)

[4.1.4 Notizen zum Auftrag 10](#_Toc188879134)

[4.1.5 Mockups zur Benutzeroberfläche der mobilen Applikation 11](#_Toc188879135)

[4.2 Anwendungsfälle Baustellen App 15](#_Toc188879136)

[4.3 Lösungsarchitektur 16](#_Toc188879137)

[5 Diskussion / Zusammenfassung 16](#_Toc188879138)

[5.1 Überschrift 16](#_Toc188879139)

[5.2 Überschrift 16](#_Toc188879140)

[Abbildungsverzeichnis 17](#_Toc188879141)

[Tabellenverzeichnis 17](#_Toc188879142)

[Glossar 17](#_Toc188879143)

[Literaturverzeichnis 18](#_Toc188879144)

[Anhang 19](#_Toc188879145)

[Selbständigkeitserklärung 20](#_Toc188879146)

# Einleitung

## Ausganslage und Motivation

Für die Versorgung ihrer Mitarbeitenden mit projektrelevanten Informationen, hat die Helion AG die Isolutions AG angefragt, eine Offerte für eine Mobile Applikation namens «THEIA Baustellen App» zu erstellen. Diese Applikation sollte mit einer API eines bestehende Systems kommunizieren. Eine erwähnenswerte Anforderung an die Applikation ist die erwartete offline Fähigkeit.

Die Helion AG ist eine Expertin für Energielösungen in der Schweiz und hat sich zum Ziel gesetzt die Energiewelt voranzutreiben. Helion AG bietet Lösungen im Bereich Photovoltaik, Stromspeicher, Smart Energy, Wärmepumpen und Ladesationen für Elektrofahrzeuge. Die Dienstleistungen umfassen die Beratung, Verkauf, Montage und Wartung und wird von einem Team von über 500 Mitarbeitenden ausgeführt.

Heute ist es besonders für die Mitarbeitenden auf der Baustelle schwierig auf die benötigten Informationen zum Projekt zu zugreifen. Ausserdem gibt es im Projektablauf Prozesse, welche die Interaktion der Mitarbeitenden erfordern. Zum Beispiel, wird für den Abschluss des Projekts ein Inbetribenahmeprotokoll mit einer vorgegebenen Checkliste verlangt. Weitere Prozesse sind Materialbestellungen, Informationsaustausch mit dem Projektmanagement oder Backoffice, usw. Heute werden diese Prozesse zum grossen Teil auf Papier durchgeführt.

Da die Anzahl der Mitarbeitenden und das Auftragsvolumen in den letzten Jahren gewachsen ist, stellt sich der Helion AG auch die Herausforderung ihre Prozesse in der Projektumsetzung zu überarbeiten. Deshalb hat sich die Helion AG dazu entschieden einen mobile Applikation für Ihre Mitarbeitenden umzusetzen.

Im Verlauf der Antragstellung dieser Master Thesis, hat sich die Helion AG dazu entschieden , die «THEIA Baustellen App» nicht in zusammenarbeit mit Isolutions zu erstellen. In Absprache mit den Experten und der Studienleitenden der Master Thesis, wurde entschieden, die Arbeit trotzdem auf Basis der Anforderungen der Helion AG durchzuführen.

## Problemstellung

Die bestehende Lösung THEIA besteht aus einem Desktop Client und einem Backendsystem. Der Kontext in dem dieses Client/Server System arbeitet, zeichnet sich durch eine stabile Netzwerkverbindung aus und der Speicher im Client ist keine limitierende Ressource. Ausserdem werden Daten höchstens lokal zwischengespeichert aber nicht persistiert, da von einem permanent verfügbaren Backen ausgegangen werden kann.

Mit der Implementierung einer mobilen Applikation werden diese zwei Vorteile der Desktop Lösung negiert. In einer mobilen Applikation kann nicht von einer durchgehenenden Netzwerkkonnektivität ausgegangen werden und der Speicher auf dem Gerät ist limitiert und meistens nicht erweiterbar. Tatsächlich ist eine spezifische Anforderung an die mobile App, die offline fähigkeit. Da gemäss Helion AG die Verbindung in das Internet auf der Baustelle oft nicht gewährleistet ist.

Aufgrund dieser Ausgangslage ist die klar, dass auf die Verfügbarkeit der Projektdaten, Funktionalität der App ohne Internet, und Konsistenz der Daten ein spezielles Augenmerk gelegt werden muss. Es gibt unterschiedliche Ansatze, Entwurfsmuster und Mehtoden für die Synchronisierung von Daten, Sicherstellung von Verfügbarkeit und Konsistenz. Es stellt sich die Frage, welches Entwurfsmuster ist am besten geeignet, um die Anforderungen der Helion AG an diese Applikation zu erfüllen.

## Ziel der Arbeit

Das Ziel dieser Arbeit ist es in einem Softwarearchitekturdokument (SAD) eine mögliche Lösungsarchitektur aufzuzeigen. Folgende Lieferobjekte werden erarbeitet, um die Fragen der Problemstellung zu beantworten.

* Lösungsarchitektur für die «Baustellen App» nach Arc42.
* Architekturentscheid für die Umsetzung der Datensynchronisation.
* ~~Minimum Viable Product (MVP) der Baustellen App umsetzen.~~

In Absprache mit dem Expertengremium wurde am 27.01.2025 entschieden, die Zielsetzung „ein MVP der Baustellen App umzusetzen“ angepasst. Neu Lautet das Ziel:

* Ein PoC der Baustellen App umsetzen.

Der PoC muss folgende Anforderungen erfüllen:

* Er zeigt die Funktion des eingesetzten Entwurfsmusters für die Datensynchronisation.
* Er demonstriert den Zugriff eines Benutzers auf Projektdaten.

## Abgrenzung

Folgende Punkte sind nicht Teil der Masterthesis:

* Die Lösungsarchitektur konzentriert sich auf die Frage der Datensychnronisation und offline Modus. Eine vollständige Implementierung der Funktionalen Anforderungen ist nicht Teil dieser Arbeit.
* Auf Grund der vorhandenen Infrastruktur beschränkt sich der PoC auf die Plattform Apple iOS.
* Eine Integration in die IT-Landschaft der Helion AG ist nicht teil dieser Masterthesis.

Wie bereits in der Ausgangslage erwähnt, hat sich die Helion AG aus dem Projekt zurückgezogen. Das führt zu einigen Einschränkungen:

* Es steht kein Backend mehr zur Verfügung. Deshalb muss für den MVP ein eigenes Backend erstellt werden.
* Auf Grund der zeitlichen Einschränkung ist Sicherheit kein Bestandteil dieser Arbeit.
* Die Lösungsarchitektur beschreibt den MVP entsprechend der Anforderungen der Helion AG.

In Absprache mit dem Expertengremium wurde am 27.01.2025 die Zielsetzung für die „Umsetzung eines MVP der Baustellen App“ angepasst. Deshalb ergeben sich zusätzliche Abgrenzungen:

* Der PoC hat keine Benutzerverwaltung. Es wird ein statischer Benutzer hinterlegt.
* Der PoC wird nicht den Funktionsumfang enthalten, wie er in der Lösungsarchitektur beschrieben wird.

## Aufbau der Arbeit

Die Masterarbeit ist folgendermassen aufgebaut. Zuerst wird in einer Einführung die Ausgangslage, Motivation, Problemstellung und die Zielsetzung erläutert.

Das zweite Kapitel beschäftigt sich mit dem Literaturreview rund um das Thema Datensynchronisation. Der Autor setzt sich mit der Beziehung und Problematik von Partitionstoleranz, Konsistenz und Verfügbarkeit in einem Verteilten System ausseinander und beleuchtet Datenübertragung, -synchronisation und -speicherung für mobile Applikationen.

Im dritten Kapitel geht der Autor auf die Methodik ein. Durch Transparenz in der Methodik soll die Reproduzierbarkeit gewährleistet werden und die Grundlage für eine kritische Bewertung legen.

Im vierten Kapitel ist das Pflichtenheft aus der Anfrage der Helion AG an die Isolutions AG aufgeführt. Dieses Pflichtenheft ist die Grundlage für die Lösungsarchitektur.

Im fünften Kapitel werden die Lieferobjekte beschrieben. Im ersten Teil wird die Lösungsarchitektur aufgezeigt. Der zweite Teil befasst sich bessonders mit der Datensynchronisation und des gewählten Entwurfsmusters. Die Validierung der Umsetzung des MVPs schliesst dieses Kapitel ab.

# Literaturreview

Dieses Kapitel beinhaltet die Grundlagen zur Datensynchronisation und Datenkonsistenz.

## Grundlagen

In einem Multiusersystem mit verteilten Komponenten, wie beispielsweise einer Webapplikation mit einer Frontendkomponente als Client und dem Backend als Server, gibt es zum einen den Aspekt der Synchronisation von Daten zwischen den Komponenten und zum anderen den simultanen Zugriff von meheren Clients auf die möglicherweise gleichen Datensätze. Der Aspekt der Datensynchronisation hat vor allem Einfluss auf die Verfügbarkeit der Anwendung und damit auf das Benutzererlebnis. Der zweite Aspekt beeinflusst primär die Konsistenz der Daten, welche für eine Geschäftsapplikation wie in diesem Fall, von grosser Bedeutung ist.

### Set Reconciliation Problem

Das «Set Reconciliation Problem» beschreibt den Umstand, dass in einem System mit verteilten Knoten, ein Datenset sich auf einem Knoten über die Zeit verändert und die Änderungen an einem bestimmten Punkt wieder zusammengeführt werden sollen [1]. Jedes Client/Server System oder ein System bestehend aus Backend/Server und mobilen Clients wird mit diesem Problem konfrontiert. Eine einfache Lösung wäre ein Reset der Datensets von einem Masterset aus [1]. Komplexere Lösungen verwenden Algorithmen um die Differenz der Datensets zu bestimmen und zusammenzuführen [2]. Primär hängt die Wahl der verwendeten Lösung ab von den Anforderungen an das System, die für die Umsetzung verfügbare Zeit und Budget und der verfügbaren Infrastruktur (Speicher, Netzwerkbandbreite, Rechenleistung) ab [1].

### CAP Theorem

Gemäss [3] spielt in einem verteilten System das CAP Theorem eine Rolle. Das CAP Theorem wurde als erstes von Eric Brewer im Kontext eines Webservices implementiert von einem Set an Servern, in geografisch verteilten Datencentren [3]. Der Einflussbereich des CAP Theorems ist anwendbar auf jedes System welches gemeinsame Datensets in verteilten Konten verwendet [3]. In den nächsten Abnschitten werden die drei Komponenten des Cap Theorems erläutert.

#### Konsistenz

Consistency bzw. Konsistenz beschreibt die Eigenschaft, dass der Informationsgehalt von Daten für alle Komponenten oder Teilnehmer in einem System zu jedem Zeitpunkt gleich ist [3].

A diagram of a server

Description automatically generated

Abbildung Konsistentes System

A diagram of a server

Description automatically generated

Abbildung Inkonsistentes System

Abbildung 7 zeigt ein verteiltes System von drei Clients, welche den Datensatz {p} vom Server abfragen. {p} ist in diesem Fall konsistent, da alle Komponenten im System mit {p1} den gleichen Informationsgehalt von {p} haben.

Abbildung 8 zeigt ein inkonsistentes System. Dies ist beispielsweise in dem Moment der Fall, wenn der Client A den Datensatz {p} bearbeitet und an den Server sendet, mit der Anweisung den Datensatz {p} anzupassen zu {p2}. Der Datensatz {p} ist nicht mehr Konsistent da der Client B und C den Datensatz {p1} haben, wärend Client A und der Server den Datensatz {p2} haben.

Um die Konsistenz wieder herzustellen, muss Client B und C den Datensatz {p} abfragen, um die Version {p2} zu erhalten.

#### Verfügbarkeit

Als Verfügbar bzw. Available gilt ein Service, wenn der Service auf eine Anfrage eine Antwort zurücksendet. Je nach Art des Systems ist die Zeitspanne von der Anfrage bis zu der Antwort auch von Bedeutung. In einem Echtzeit-System oder einer Benutzeranwendung ist eine Antwort welche zu lange braucht genau so schlecht wie keine Antwort [3].

#### Partitionstoleranz

Die Partionstoleranz bzw. Partition Tolerance beschreibt die Fähigkeit eines verteilten Systems, mit Kommunikationsausfällen oder -verzögerungen umzugehen. Ein System ist partitionstolerant, wenn es trotz Kommunikationsausfall zwischen Knoten im System weiter arbeiten kann [4].

## Entwurfsmuster für die Datensynchronisation in mobilen Anwendungen

Gemäss [5] gibt es einige Design Pattern für die synchronisation von Daten im mobilen Applikationsdesign. Diese Entwurfsmuster werden in diesem Abschnitt beschrieben. Sie lassen sich grob in 3 Bereiche aufteilen, „Datensynchronisation“, „Datenspeicher und Verfügbarkeit“ und „Datenübertragung“ [6].

### Datensynchronisation

Dieser Bereich lässt sich in zwei Entwurfsmuster einteilen. Synchrone und Asynchrone Datensynchronisation. Beide Varianten haben Vor- und Nachteile und eignen sich für spezifische Einsatzgebiete.

#### Synchrone Datenübertragung

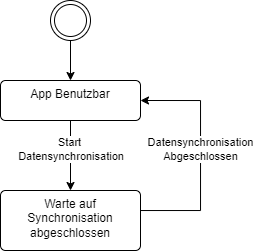


Abbildung Datensynchronisation als synchroner Fluss [5]

In Abbildung 3 wird ein synchroner Datenfluss in einer mobilen App dargestellt. Eine Synchrone Datenübertragung führt in jedem Softwaresystem dazu, dass der Prozess der Anfragenden Komponente auf die Antwort wartet. Im Kontext einer Mobilen Applikation bedeutet dies, dass die Benutzeroberfläche blockiert wird. Erst wenn die Synchronisation der Daten abgeschlossen ist, kehrt die Applikation wieder in einen benutzbaren Zustand zurück. Die synchrone Datenübertragung kommt in folgenden Fällen zur Anwendung:

* Ein komplettes Datenset ist für die Funktion der Applikation dringend nötig [5].
* Wenn die Antwort nötig ist um zu bestimmen in welchen Status die Applikation vorrücken soll [5].

**Vorteile:**

* Das Statusmanagement der Anwendung ist simpel. Da der Statuswechsel strikt seqeunziell ist, werden zusätzliche Zustände, wie sie in der asynchronen Datenübertragung verwendet werden, nicht benötigt [5].

**Nachteil:**

* Wie bereits in der Erklärung beschreiben, ist die Benutzeroberfläche für den Benutzer gesperrt, weil der darunterliegen Prozess auf die Anwtort der Datensynchronisation wartet. Entsprechend leidet die Benutzererfahrung [5].

#### Asynchrone Datenübertragung

A screenshot of a phone

Description automatically generated

Abbildung Darstellung einer asynchronen Datensynchronisation

In der Abbildung 10 ist der Abblauf einer asynchronen Datensynchronisation zu sehen. Der Unterschied zur synchronen Datensynchronisation ist, dass sich die Applikation bei der Initiierung der Synchronisation in zwei verschiedene Zustände begibt. Die Benutzeroberfläche kehrt in den benutzbaren Zustand zurück, wärend gleichzeitig die Datensynchronisation im Hintergrund läuft [5] dargestellt als Zustand „Synchronisieren“.

**Vorteil:**

* Die Applikation bleibt für den Benutzer benutzbar [5].
* Ausserdem erlaubt eine asynchrone Datensynchronisation den Austausch von Daten auch wenn die mobile App nicht im Vordergrund ist [5].

**Nachteile:**

* Mehrere konkurrierende Anfragen an ein Datenset können auftreten und müssen behandelt werden [5]. Eine mögliche Lösung ist die asynchrone Übertragung von Transaktionen, welche nach der Übertragung synchron angewendet werden [5].
* Die übertragene Datenmenge kann nicht vorausgesagt werden. Das kann einen negativen Einfluss auf die Kosten der Netzwerkbandbreite und deren Verfügbarkeit haben [5].

### Datenspeicher und Verfügbarkeit

In diesem Bereich gibt es ebenfalls zwei Entwurfsmuster, welche sich auf die lokale Verfügbarkeit auf dem Gerät auswirken. Bei der Wahl dieser Entwurfsmuster sind bessonders der verfügbare Speicherplatz auf dem Gerät und die Netzwerkbandbreite von Bedeutung [5].

#### Parial Storage

Das erste Entwurfsmuster ist „Partial Storage“ bzw Teilspeicher. Dieses Entwurfsmuster reduziert den Speicherbedarf der Daten auf dem Gerät, da mit diesem Muster nur die gerade benötigten Daten vom Backend geladen werden.

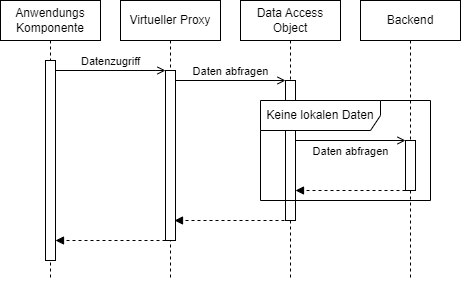


Abbildung Ablauf von Partial Storage Datenabfragen

In Abbildung 5 ist der Ablauf und das Prinzip von „Partial Storage“ zu sehen. Oft wird das Muster „Partial Storage“ mit der Implementierung des „Virtual Proxy“ und des „Data Access Object“ Entwurfmusters umgesetzt [5]. Der „Virtual Proxy“ imitiert das eigentliche Datenobjekt, in dem es das gleiche Interface wie das Datenobjekt implementiert [7]. Der virtuelle Proxy fängt Methoden Calls und Zugriffe auf Objekt Attribute ab und erlaubt so eine gezielte Initialisierung von Feldern in dem Moment wenn sie gebraucht werden, dieses Vorgehen nennt sich auf „Lazy Loading“ [5]. Das „Data Access Object“ bzw. DAO ist für die Abfrage der Daten zuständig [7] in dem es den Zugriff auf Daten abstrahiert. Sind die angefragten Daten im lokalen Speicher oder einem Cache verfügbar, antwortet das DAO mit einer Kopie der Daten aus dem lokalen Speicher. Ansonsten initiert das DAO eine Datenabfrage über das Netzwerk an das Backend [5].

**Vorteile:**

* Der benötigte Speicher auf den mobilen Gerät ist reduziert, da nur die benötigten Teile eines Datensatzes in den lokalen Speicher geladen werden [5]. Dieser Umstand erlaubt auch die Verwendung von weitaus umfangreicheren Datensätzen [5].
* Dieses Muster erlaubt eine granulare Kontrolle darüber, wann welche Daten synchronisiert werden [5].

**Nachteile:**

* Die Netzwerkverbindung ist für dieses Muster ein Schwachpunkt. Da das Lazy-Loading eine Netzwerkverbindung erfordert um Daten nachzuladen, kann ein Unterbruch des Netzwerks die mobile App unbrauchbar machen [5].

#### Complete Storage

Das zweite Entwurfsmuster ist der Complete Storage bzw. Komplettspeicher. Im Gegensatz zum Partial Storage wird in diesem Muster der komplete Datensatz auf ein mobiles Gerät geladen. Dieser Prozess wird üblicherweise beim Start der Applikation ausgeführt, damit subsequente Datenanfragen immer aus dem lokalen Speicher beantwortet werden können.

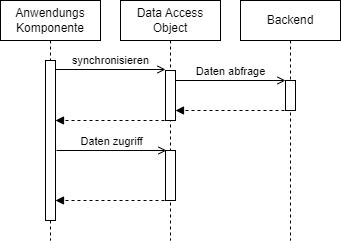


Abbildung Darstellung der Complete Storage Datenabfrage

In der Abbildung 6 ist zu erkennen, dass der Datentransfer initial gestartet wird und das DAO alle Datensätze aus dem Backend abfragt. Die Daten werden im lokalen Gerätespeicher abgelegt und subsequente Datenzugriffe erfolgen aus dem lokalen Speicher [5].

**Vorteil:**

* Die Abhängigkeit zur Netzwerkkonnektivität ist gering. Nachdem die Daten initial synchronisiert wurden, kann die Applikation auch ohne Konnektivität verwendet werden. [5]

**Nachteil:**

* Es wird mehr Specher auf dem mobilen Gerät vorausgesetzt [5].
* Die Datenübertragung ist verschwenderisch was die Netzwerkbandbreite anbelangt. Weil beim Synchronisationsereignis alle Daten auf einemal übertragen werden [5].

### Datenübertragung

Dieser Abschnitt behandelt das Thema Datenübertragung. Das Ziel ist die optimierte Übertragung von Daten aus dem Backend zu der mobilen Applikation und zurück. Bei der Wahl des Entwurfmuster zur Datenübertragung, muss primär der Typ der zu übertragenden Daten, die verfügbare Netzwerkbandbreite in Betracht gezogen werden [5].

#### Full Transfer

Für kleine Datensätze oder Daten, welche jeweils als ganzes verändert werden, beispielsweise ganze Dateien, eignet sich das Entwurfsmuster „Full Transfer“. Das Entwurfsmuster zeichnet sich dadurch aus, dass der komplete Datensatz übertragen wird und den aktuellen Datensatz auf dem mobilen Gerät ersetzt.

A black background with white squares

Description automatically generated

Abbildung Darstellung des Full Transfer Übertragungsmuster

In der Abbildung 7 ist zu erkennen, dass der „Full Transfer“ ein sehr einfaches Entwurfsmuster ist. Der Transfer wird für das gesamte Datenset initiert [5].

**Vorteile:**

* „Full Transfer“ hat den Vorteil, dass es ein einfach zu implementierendes Entwurfsmuster ist [5].

**Nachteile:**

* Der Nachteil ist, dass die Daten redundant gesendet werden und dadurch wird Bandbreite verschwendet [5].

#### Timestamp Transfer

Das Muster „Timestamp Transfer“ basiert auf dem Vergleich von Zeitstempeln in einem Datensatz.

A screenshot of a cell phone

Description automatically generated

Abbildung Darstellung des Musters Timestamp Transfer

Die Abbildung 7 ist dargestellt, wie die prinzipielle Logik eines „Timestamp Transfers“ aussieht. Damit dieser Vergleich möglich ist, muss der Datensatz mit zusätzlichen Felder mit dem Zeitstempel der letzten Mutation und der Erstellung des Datensatzes erweitert werden [5]. Die mobile Applikation sendet die Zeitstempel an den Server. Dieser vergliecht die Zeitstempel von der mobilen Applikation mit den eigenen Datensätzen und sendet aktuellere Datensätze an den Client [5]. Es werden jeweils ganze Datensätze übermittelt.

**Vorteile:**

* Die zu synchronisierenden Daten können gezielt anhand ihrer aktualität ausgewählt werden. Sowohl für den Upload wie auch für den Download. Dadurch wird die Menge an zu übertragenden Daten verringert [5].

**Nachteile:**

* Die Löschung von Datensätzen benötigt zusätzliche Logik. Da in diesem Fall der gelöschte Datensatz nicht mehr für den Zeitstempelvergleich zur Verfügung steht[5].
* Die Quelle für den Zeitstempel muss konsistent sein, damit die Zeitstempel vergleichbar bleiben [5].

#### Mathematical Transfer

Das dritte Entwurfsmuster macht sich mathematische Algorithmen zu Nutze, um zu bestimmen welche Teile eines Datensatzes übermittelt werden sollen [5]. Es werden nur Daten ausgetauscht, welche geändert wurden. Die mathematischen Algorithmen sind in der Lage, Änderungen effizienter zusammen zu führen und die Menge an Daten in einer Übertragung auf ein minimum zu reduzieren [5]. Beispiele für mathematische Methoden sind:

* Der Vergleich von Checksummen [5]
* „Invertible Bloom Lookup Tables“ (IBLTs) [8]
* “Characteristic Polynomial Interpolation” [5]

A screenshot of a computer screen

Description automatically generated

Abbildung Darstellung des Musters Mathematischer Transfer

Wie in Abbildung 9 zu sehen ist, weist das Muster „mathematischer Transfer“ Ähnlichkeiten zum „Timestamp Transfer“ Muster auf. Der Unterschied ist die mathematische Berechnung der Änderungen [5].

Vorteile:

* Mit diesem Muster kann die benötigte Bandbreite auf ein Minimum reduziert werden [5]. Das Muster kommt in Frage, wenn die effiziente Nutzung der verfügbaren Bandbreite kritisch ist.

Nachteile:

* Mathematische methoden für die Berechnung sind oft stark abhängig vom Kontext der Daten [5]. Ändert der Datenkontext muss auch der Algorithmus angepasst werden.
* Die Komplexität von Mathematischen Methoden ist hoch. Entsprechend wird für die Implementierung mehr Zeit benötigt [5].

# Methodik

In diesem Kapitel wird die verwendete Methodik beschrieben Durch die Transparenz in der Methodik wird nicht nur die Reproduzierbarkeit gewährleistet, sondern auch die Grundlage für eine kritische Bewertung der Arbeit gelegt.

## Projektmanagement

Für die Umsetzung wurde eine Agile Vorgehensweise gewählt. In einem Projektboard in Github werden Tasks erfasst und iterativ abgearbeitet. Die Tasks sind primär auf die Lieferobjekte zugeschnitten. Das Projektboard orientiert sich dabei an der Kanban Methode [9]. Das Projektboard befindet sich auf Github:

<https://github.com/users/zuercheram/projects/2/views/3>

Der Gesamtvortschritt der Masterthesis ist in einem übergeordneten Projektplan dargestellt. Er zeigt die Termine, Meilensteine und geplante Umsetzungsabschnitte für die Masterthesis.

## Masterthesis Zeitplan



## Entwurf der Lösungsarchitektur

## Analyse der Datensynchronisation

## Umsetzung der Baustellen App als MVP

# Ergebnisse

In diesem Kapitel werden die Ergebnisse der Arbeit dargelegt. Zuerst werden die Anforderungen an die „Baustellen App“ dargelegt (Kapitel 4.1). Anschliessend folgen die Lösungsarchitektur für den MVP (Kapitel 4.3) inklusive der Nutzertanalyse (Kapitel 4.3.7.4). Zum Abschluss wird die Implementierung beschrieben (Kapitel ).

## Pflichtenheft

Im Pflichtenheft sind die buisnessrelevanten Anforderungen an die Applikation aufgeführt. Zusätzlich zu den Anforderungen sind Mockups im Pflichtenheft enthalten. Die Mockups visualisieren wie Anforderungen in die Benutzeroberfläche realisiert werden sollen. Die Mockups sind Illustrationen und fokusieren sich auf das Layout und positionierung von Informationen auf dem Bildschirm. Je nach Plattform werden sich die Elemente visuell von den Mockups unterscheiden.

Die Anforderungen und Mockups sind aus dem Angebot «Konzeption THEIA für die Baustelle» für die Helion AG abgeleitet. Das Angebotsdokument wurde von der Isolutions AG verfasst. Da das Angebotsdokument interne Informationen enthält, wurden die Anforderungen generalisiert.

Die Helion AG will die «Baustellen App» mehreren Phasen umsetzen. Die erste Implementierung soll als MVP ausgelegt sein. Mit dieser ersten Version soll die Stossrichtung der mobilen Applikation und die Akzeptanz bei den Anwendenden überprüft werden. Weitere Funktionen werden anschliessend geplant und implementiert.

### Authentifizierung

In diesem Abschnitt sind die Anforderungen an die Authentifizierung aufgelistet.

Tabelle Anforderungen an die Authentifizierung

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| REQ-001 | Authentisierung | Die Benutzer müssen sich beim Einstieg in die App als erstes authentifizieren. Erst nach Authentisierung und Autorisierung können die Benutzer auf die Inhalte zugreifen. |
| REQ-002 | Speicherung der Session | Nach dem Einloggen bleibt die Session des Benutzers in der App gespeichert, damit sich der Benutzer nicht immer wieder neu anmelden muss. Nach einer zu definierenden Zeitdauer wird die Session ungültig und der Benutzer muss sich neu anmelden. |
| REQ-003 | Logout | Der Benutzer muss die Möglichkeit haben, sich wieder abzumelden. |

### Projektübersicht

In diesem Abschnitt sind Anforderungen an die Projektübersicht.

Tabelle Anforderungen an die Projektübersicht

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| REQ-004 | Projektliste | Nach dem Start der Applikation soll als erste Seite eine Liste mit den Projekten dargestellt werden. Vorausgesetzt der Benutzer ist eingeloggt. |
| REQ-005 | Personalisierte Projekte | In der Projektliste werden nur diejenigen Projekte angezeigt, auf die der Benutzer die Berechtigung zum Zugriff hat. |
| REQ-006 | Projektname | Jedes Projekt ist mit dem Namen und der Ortschaft der Baustelle aufgelistet. |
| REQ-007 | Projektsuche | In einem Suchfeld kann ein Projekt nach dem Namen und der Ortschaft gesucht werden. Die Liste passt sich entsprechend dem Suchbegriff laufend an. Der Suchbegriff muss wieder gelöscht werden können. |
| REQ-008 | Projektauswahl | Mittels Touch auf eines der Projekte können die Projektdetails aufgerufen werden. |
| REQ-009 | Offlinefähigkeit | Projekte sollen auch Offline verfügbar sein. Der Benutzer kann in der Projektliste Projekte markieren welche offline verfügbar sein sollen. Sämtliche Projektdaten zu einem markierten Projekt sollen offline verfügbar sein. |

### Projektdetails

In diesem Abschnitt sind Anforderungen an die Projektdetails.

Tabelle Anforderungen an die Projektdetails

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| REQ-010 | Projektdetails | Es werden die wichtigsten Infos eines Projektes gemäss obigen Mockups angezeigt. Ist eine Internetverbindung vorhanden, werden die Daten vom Backend geladen. Ohne Internetverbindung werden die Daten aus dem lokal Speicher geladen. |
| REQ-011 | Adressen in Maps öffnen | Kontakt- und Objektadresse können per Touch in Karten (ios) oder Maps (android) geöffnet werden. |
| REQ-012 | Absprung auf externe Ablagen | Wird auf dem Projekt ein Link zu einer externen Ablage mitgeliefert, wird in der Detailsansicht ein Link Button auf die externe Ablage angezeigt. |
| REQ-013 | Notizen zum Auftrag | Über einen weiteren Button kann die Projektnotizen-Funktion aufgerufen werden. |
| REQ-014 | Inbetriebnahme | Über einen weiteren Button kann die Inbetriebnahme-Funktion aufgerufen werden. |
| REQ-015 | Zurück zur Übersicht | Mittels eines Zurück-Buttons muss wieder in die Projektübersicht gewechselt werden können. |
| REQ-016 | Offlinefähigkeit | Sämtliche Daten dieses Projekts müssen auch ohne aktive Internetverbindung auf den Smartphones abrufbar sein. Online Links auf externe Ablagen funktionieren im Offline-Modus nicht. |
| REQ-017 | Absprung nach MS Teams | In den Projekten kann ein Link auf einen Teams Channel oder Chat hinterlegt werden. In der Projektansicht wird dieser Link als Button angezeigt. Ein Klick auf den Button, leitet den Benutzer weiter zu Teams. |
| REQ-018 | Absprung nach MS SharePoint | In den Projekten kann ein Link auf eine SharePoint Ablage hinterlegt werden. In der Projektansicht wird dieser Link als Button angezeigt. Ein Klick auf den Button leitet den Benutzer weiter auf die SharePoint Ablage. |
| REQ-019 | Logistik Fehler melden | Logistik Fehler werden in einer bestehenen Power App erfasst. In der Projektansicht gibt es einen Button mit einem Deeplink auf diese Power App. Ein Klick auf den Button leitet den Benutzer zur Power App für die Meldung von Logistik Fehlern. |

### Notizen zum Projekt

In diesem Abschnitt werden Anforderungen zu den Projektnotizen aufgeführt.

Tabelle Anforderung zu den Notizen

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| REQ-020 | Notizen-Übersicht | In einer Liste werden sämtliche bisher erfassten Notizen zum ausgewählten Projekt angezeigt. |
| REQ-021 | Filter nach Kategorie | In einem Dropdown Feld kann die jeweilige Notizen-Kategorie gefiltert werden. Standardmässig ist kein Filter gesetzt. |
| REQ-022 | Detailansicht Notizen | In einer Detailansicht können die Notizen angezeigt werden. Externe Links (z.B. auf Word-Dokumente welche in SharePoint abgelegt sind) können angeklickt und im externen Browser (Safari oder Chrome) angezeigt werden, falls die entsprechenden Berechtigungen vorhanden sind. Eingebettete Bilder werden geladen und angezeigt. |
| REQ-023 | Notiz editieren und erstellen. | Notizen können editiert und erstellt werden. |
| REQ-024 | Zurück zur Übersicht | Mittels eines Zurück-Buttons muss wieder in die Notizen-Übersicht gewechselt werden können. |
| REQ-025 | Offlinefähigkeit | Sämtliche Notizen der geladenen Projekte müssen auch ohne aktive Internetverbindung auf den Smartphones abrufbar sein. Externe Links funktionieren offline nicht. |

### Inbetriebnahme

In diesem Abschnitt werden Anforderungen zu der Inbetriebnahme aufgeführt.

Tabelle Anforderungen zu der Inbetriebnahme

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| REQ-026 | Inbetriebnahme | Für jedes Projekt kann eine Inbetriebnahme erfasst werden. Ist bereits eine Inbetriebnahme da, kann diese editiert werden. |
| REQ-027 | Checkliste anzeigen | Es werden drei Checklisten mit erforderlichen Dokumenten, IBN SOP 2.0 und Bilder angezeigt. Die Checkpunkte können je Projekt verschieden sein. Jedes Checklisten-Item ist entweder Pflichtfeld oder Optional. |
| REQ-028 | Vorhandene Bilder | Sind bereits Bilder vorhanden, können diese via Ansicht-Icon angezeigt werden (aktuell nicht vorhanden im Mockup) |
| REQ-029 | Neue Bilder hinzufügen | Im App kann pro Checklisten-Item eines oder mehrere Bilder Heraufgeladen werden. Entweder werden diese gleich mit der Kamera aufgenommen, oder sie werden aus der Camera Roll hinzugefügt. Sobald die Fotos hochgeladen wurden, sind sie auch über die Ansicht verfügbar (REQ-28). |
| REQ-030 | Anmerkungen | Es können Anmerkungen in einem Textfeld erfasst oder editiert werden. Dabei ist keine Formatierung vorgesehen, also einfach ein Textfeld. |
| REQ-031 | Speichern und Inbetriebnahme | Der aktuelle Stand der Checkliste kann jederzeit gespeichert oder gespeichert und abgeschlossen werden. Ist der IBN einmal abgeschlossen, kann er im App nur noch angeschaut werden. |
| REQ-032 | Offlinefähigkeit | Sämtliche Inbetriebnahmen der geladenen Projekte (eines je Projekt) müssen auch ohne aktive Internetverbindung auf den Smartphones abrufbar sein. Auch müssen vorhandene Bilder angezeigt und neue Bilder (Multi-upload) hinzugefügt werden können. Sobald eine Onlineverbindung wieder verfügbar ist, werden die Daten über die APIs ins Backend geladen. |

### Mockups zur Benutzeroberfläche der mobilen Applikation

Mit den Mockups soll das Konzept der Benutzeroberfläche visualisiert werden. Die Mockups wurden mit dem UI/UX Design Tool Figma[[1]](#footnote-1) erstellt. Benutzt wurden standart Asset Bibliotheken. Das Look and Feel der visuellen Elementen entspricht nicht dem finalen Aussehen in der Applikation, sondern dient nur zur Veranschaulichung des Benutzeroberfläche und der User Experience

#### Benutzeroberfläche für die Authentifizierung

A screenshot of a login screen

Description automatically generated

Abbildung Loginmaske

#### Projektbezogene Ansichten

A screenshot of a phone

Description automatically generated

Abbildung Projektübersicht

A screenshot of a phone

Description automatically generated

Abbildung Projektdetailansicht

#### Notizenansichten

A screenshot of a phone

Description automatically generated

Abbildung Notizenübersicht

A screenshot of a cell phone

Description automatically generated

Abbildung Notizdetailansicht

A screenshot of a phone

Description automatically generated

Abbildung Neue Notiz erfassen

### Nicht Funktionale Anforderungen

In diesem Abschnitt werden die nicht funktionalen Anforderungen augelistet

Tabelle Auflistung der nicht funktionalen Anforderungen

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| REQ-NF-001 | Leistung | Die App soll schnell und reaktionsschnell sein, insbesondere beim Laden von Inhalten und beim Wechseln zwischen den Ansichten. |
| REG-NF-002 | Benutzerfreundlichkeit | Die Benutzeroberfläche sollte intuitiv gestaltet sein, um eine nahtlose Experience zu gewährleisten. |
| REG-NF-003 | Sicherheit | Die App sollte sicher sein, sowohl in Bezug auf Datenübertragung wie auch lokale Datenspeicherung. Authentifizierung über Azure AD muss möglich sein. |
| REG-NF-004 | Verteilbarkeit | Die App sollte möglichst einfach auf den Geräten der Mitarbeitenden installiert werden können. |
| REG-NF-005 | Wartbarkeit | Die App sollte wartbar sein, neue Versionen sollen möglichst einfach auf die Geräte der Mitarbeitenden verteilt werden können. |
| REG-NF-006 | Offlinefähigkeit | Damit die App auch auf der Baustelle und im Keller verwendet werden kann, sollen die Funktionen auch offline verfügbar sein. Nicht geladene Inhalte sollen bei Onlineverbindung heruntergeladen werden können. Für Online-only Inhalte sollte eine passende Fehlermeldung angezeigt werden. |
| REG-NF-007 | Gerätekompatibilität | Die App soll auf Android Version 12 und neuer sowie iOS 16 und neuer lauffähig sein. |

## Anwendungsfälle Baustellen App

Basierend auf den Anforderungen aus dem Pflichtenheft, wurden für den MVP der Baustellen App Anwendungsfälle abgeleitet. Die Anwendungsfälle dienen zur Visualisierung welche Aktionen das System asuführen kann, welche Aktoren mit dem System interagieren und in welche Beziehung die Aktoren und Aktionen stehen. [10]

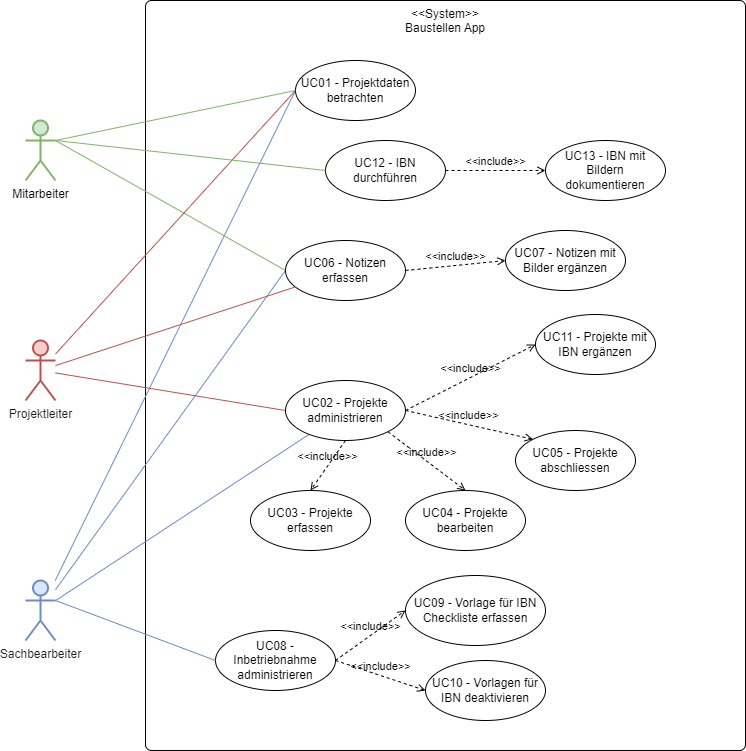


Abbildung Anwendungsfalldiagramm für die Applikation Baustellen App

In der nachfolgenden Tabelle sind die Anwendungsfälle beschrieben.

Tabelle Beschreibung der Anwendungsfälle

|  |  |  |
| --- | --- | --- |
| ID | Name | Kurzbeschreibung |
| UC01 | Projektdaten betrachten | Projektdaten können von allen Akteuren der «Baustellen App» betrachtet werden. Sachbearbeiter betrachten Projekte in der «Baustellen App» via Desktop Client. Projektleiter und Mitarbeiter müssen Projekte mit dem moblien Client aufrufen können. |
| UC02 | Projekte administrieren | Sachbearbeiter und Projektleiter administrieren die «Baustellen App» Projekte. |
| UC03 | Projekte erfassen | Sachbearbeiter und Projektleiter eröffnen Projekte. |
| UC04 | Projekte bearbeiten | Sachbearbeiter und Projektleiter bearbeiten Projekte |
| UC05 | Projekte abschliessen | Sachbearbeiter und Projektleiter schliessen Projekte ab. |
| UC06 | Notizen erfassen | Sachbearbeiter, Projektleiter und Mitarbeiter können Notizen zu einem Projekt erfassen. Diese Notizen erlauben den Austausch von zusätzlichen, projektbezogenen Information. |
| UC07 | Notizen mit Bildern ergänzen | Sacbearbieter, Projektleiter und Mitarbeiter können zu ihren Notizen Bilder hinzufügen. Die Bilder stammen entweder aus der Fotobibliothek des Gerätes oder werden direkt mit der Gerätekamera aufgezeichnet. |
| UC08 | Inbetriebnahme (IBN) administrieren | Sachbearbeiter verwalten die Vorlagen für die Checklisten. |
| UC09 | Vorlage für IBN Checkliste erfassen | Sachbearbeiter erfassen neue Vorlagen für Checklisten. |
| UC10 | Vorlage für IBN deaktivieren | Sachbearbeiter deaktivieren Vorlagen, damit diese nicht weiter verwendet werden können. |
| UC11 | Projekte mit IBN ergänzen | Sachbearbeiter oder Projektleiter wählen aus den Vorlagen der IBN Checklisten eine aus und fügen diese Vorlage einem Projekt hinzu. |
| UC12 | IBN durchführen | Mitarbeiter führen die IBN auf der Baustelle durch und füllen dabei die Checkliste in der «Baustellen App» aus. |
| UC13 | IBN mit Bildern dokumentieren | Für die Belegbarkeit der durchgeführten IBN erfasst der Mitarbeiter Bilder und fügt diese Bilder der IBN hinzu. |

## Lösungsarchitektur

In diesem Kapitel wird die Lösungsarchitektur für die Baustellen App aufgezeigt. Als Grundlage für die Dokumentation wurde das Template Arc42 von Dr. Peter Hruschka und Gernot Starke verwendet [11]. In der Lösungsarchitektur wird die Umsetzung der Baustellen App als MVP bezeichnet. Im Rahmen der Masterthesis wird die Baustellen App als PoC umgesetzt, um spezifische Anwendungsfälle aufzuzeigen. Die Lösungsarchitektur hingegen beschreibt den kompletten Funktionsumfang des MVPs.

### Einführung und Ziele

Die «Baustellen App» ist ein Softwaresystem und soll als Kernanwendung für die Projektabwicklung der Helion AG dienen. Von der Beratung über den Verkauf, Planung und Durchführung der Dienstleistungen der Helion AG, bietet die Baustellen App Unterstützung für die durchführenden Personen und dient als zentraler Zugriffspunkt für Projektdaten, Administration und Prozessdurchführung. In die Baustellen App soll in einem ersten Schritt als MVP mit einem minimalen Funktionsumfang für eine kleine Benutzergruppe umgesetzt werden.

### Qualitätsziele

Die Qualitätsziele an den MVP sind aus den Anforderungen der Helion AG abgeleitet. Im Zentrum steht die Prämisse: Die Lösung soll die Mitarbeitenden in ihrer Arbeit unterstützen.

Tabelle Qualitätsanforderungen an die Baustellen App

|  |  |  |
| --- | --- | --- |
| ID | Qualitätsziel | Beschreibung |
| QZ-001 | **Zuverlässigkeit**   * Verfügbarkeit | Ein wichtiges Qualitätsziel ist die Verfügbarkeit der App und der Daten in der App. Es ist dringend notwendig, dass der Monteur auf der Baustelle zugang zu den Projektdaten hat und wichtige Arbeitsschritte wie die Materialbestellung und Inbetriebnahme durchführen kann. Dies muss er auch tun können, wenn keine Internetverbindung vorhanden ist. |
| QZ-002 | **Sicherheit**   * Integrität * Nachweisbarkeit | Die zentrale Speicherort für die Daten der Applikation, ist das Backend System der Baustellen App. Da wärend der Verwendung der mobilen App im offline Modus konkurierende Datenmanipulationen auftreten können, muss sichergestellt werden, dass die Integrität der Daten gewährleistet ist.  Das System muss Konflikte in konkurrierenden Datenbearbeitungen transparent lösen. |
| QZ-003 | **Benutzbarkeit**   * Bedienbarkeit | Das System soll dem Mitarbeiter auf der Baustelle unterstützen. Jede Interaktion mit der Applikation muss effizient und einfach erledigt werden können. Systemprozesse sollen den Benutzer nicht blockieren.  Wenn das System auf die Antwort eines Prozesses warten MUSS, dann ist dies dem Benutzer transparent darzustellen. |

### Stakeholder

In der folgenden Tabelle werden die Stakeholder des MVPs aufgelistet und beschrieben.

Tabelle Stakeholder des Baustellen App MVPs

|  |  |
| --- | --- |
| Wer | Interesse, Bezug |
| Mitarbeiter und Mitarbeiterinnen auf der Baustelle | Will schnell und immer auf Projektdaten zugreifen und Notizen sehen und erfassen können. Erwartet Zugriff auf Funktionen und Daten via mobilem Endgerät.   * Kann alle Projekte betrachten * Kann in allen Projekten Notizen erfassen * Kann Dokumente abfragen * Kann Bilder aufnehmen und in Notizen Ablegen |
| Projektleiter/innen | Muss auf die Projektdaten zugreifen können. Sowohl im Backoffice wie auch auf der Baustelle. Will Projekte primär am Desktop administrieren. Will Projektdaten auf der Baustelle mit einem mobilen Endgerät betrachten.   * Kann Projekte erstellen * Kann **seine** Projekte bearbeiten * Kann Notizen erfassen |
| Sachbearbeiter/innen | Bearbeitet Projektdaten aus dem Backoffice. Verwendet primär einen Desktoprechner.   * Kann Projekte erstellen * Kann alle Projekte bearbeiten * Kann Notizen erfassen |

### Randbedingungen

In der folgenden Tabelle sind die relevanten Randbedingungen beschrieben.

Tabelle Randbedingungen

|  |  |
| --- | --- |
| Id | Randbedingung |
| RB001 | Der MVP muss mobile Endgeräte mit Betriebsystem ab iOS Version 16 und Android Version 12 kompatibel sein. |
| RB002 | Für die Umsetzung müssen Microsoft Technologien wie ASP.NET und C# verwendet werden. |
| RB003 | Als Plattform für die Baustellen App soll die Public Cloud von Micorosoft «Azure» zum Einsatz kommen. |

### Kontextabgrenzung

Im folgenden Abschnitt wird der Kontext der Baustellen App als Blackbox aufgezeigt und Beschrieben.

#### Fachlicher Kontext (Blackbox)

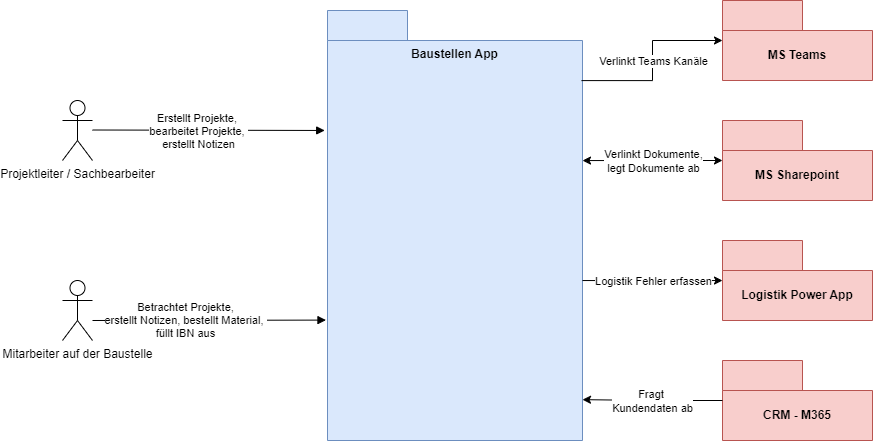


Abbildung Fachlicher Kontext

In der Folgenden Tabelle werden die Akteure/Akteurinnen und externen Systeme beschrieben.

Tabelle Akteure/Akteurinnen und System im Kontext der Baustellen App

|  |  |
| --- | --- |
| Aktor/System | Beschreibung |
| Projektleiter | Der/Die Projektleiter/in ist verwantwortlich für das Mangement der Projekte der Helion AG. Er/Sie kann die Ihm/Ihr zugewiesenen Projekte bearbeiten. |
| Sachbearbeiter | Der/ Sachbearbeiter/in ist im Backoffice der Helion AG tätig und kann alle Projekte betrachten und bearbeiten. Er/Sie mutiert Projektdaten auf Anweisung und führt im Rahmen der Backofficetätigkeit nötige Prozessschritte in der Baustellen App aus. |
| Mitarbeiter auf der Baustelle | Der/Die Mitarbeiter/in auf der Baustelle ist für die Umsetzung der Projekte auf der Baustelle zuständig. Er/Sie führt Materiallbestellungen aus und nimmt Anlagen gemäss Protokoll in Betrieb. |
| MS Teams | Wird für die Kollaboration von Personal und Kunden verwendet. Für das Projekt wichtige Chats und Kanäle, können in der Baustellen App verlinkt werden. |
| MS Sharepoint | Sharepoint wird für die Ablage von Dokumenten verwendet. Die Baustellen verlinkt einerseits auf Dokumente in Sharepoint und legt andererseits Dokumente in Sharepoint ab. |
| Logistik Power App | In diesem System werden Materialbestellungen für ein Projekt abgewickelt. Ist die Materialbestellung fehlerhaft, kann dies via «Baustellen App» gemeldet werden. |
| CRM – M365 | In diesem System werden die Kundendaten der Helion AG verwaltet. Die Baustellen App Bezieht Kundendaten aus diesem System. |

### Lösungsstrategie

In diesem Abschnitt wird die Lösungsstrategie für die «Baustellen App» aufgezeigt

#### Offline Verfügbarkeit

Um dem Mitarbeiter auf der Baustelle eine einfache Möglichkeit zu bieten auf die Projektdaten zuzugreifen, wird eine native, mobile Applikation erstellt. Dieser mobile Client bietet die Möglichkeit die Applikation offline zu nutzen, Daten zu erfassen und Bilder zu speichern. Die Datenmanipulationen werden auf dem mobilen Gerät zwischen gespeichert und in das Backend synchronisiert, wenn Konnektivität wiederhergestellt wurde.

#### Datensynchronisation

Da hier ein Multi-User System mit verteilten Komponenten und offline Fähigkeit realisiert werden soll, wird an die Synchronisation der Daten besondere Anfordrungen gestellt. Anhand einer Nutzwertanalyse (Kapitel 4.3.7.4) wurde die geeignetste Methode festgestellt. Die gewählte Methode hat die Konsequenz, dass Konflikte anhand der «Last Write Wins» Regel gelöst werden. Diese Entscheidung und damit die Konsequenz wurde von der Helion AG akzeptiert.

#### Wartbarkeit und Erweiterbarkeit

Die «Baustellen App» wird in Modular aufgebaut. Die Module sind nach fachlichen Anforderungen getrennt. Ein Modul bündelt zusammenhängende Funktionen. Die Module werden als Microservices aufgebaut. Damit wird die zukünftige Erweiterbarkeit garantiert.

#### Sicherheit

Benutzer müssen sich Authentifizieren. Auf Grund der Wahl der Helion AG der Azure Cloud als Plattform für die «Baustellen App», wird für die Authentifizierung wird Azure Entra ID eingesetzt. Ein Rollenmodell, welche entlang der fachlichen Akteure aufgebaut ist, gewährt gezielte Rechte für die Anwender.

#### Performance

Um schnelle Antwortzeiten für Anfragen an das Backend zu erreichen, wird das Backend als Microservice Architektur aufgebaut. Die Last wird so auf mehrere Serivces verteilt, welche einzeln skalierbar sind. Dadurch kann die «Baustellen App» sehr flexibel auf Lastschwankungen reagieren.

#### Web Frontend

Für das Management der Projekte wird ein Webportal realisiert. Da diese Arbeit vorwiegend am Desktop geschieht, bietete ein Webportal die bessere Benutzererfahrung für die Administration von Datensätzen als eine mobile App. Das Webportal wird als Single Page Application (SPA) mit dem bewährten Angular Framework realisiert.

#### Mobile App

Für die Anwendung im Feld kommt eine mobile App für iOS und Android zum Einsatz. In diesem Kontext ist Verfügbarkeit (auch offline) und Geschwindigkeit von Bedeutung. Die mobile Applikation ist auf diese Qualitätsmerkmale zugeschnitten, damit der Mitarbeiter auf der Baustelle die bestmögliche Effizienz in der Verwendung der Baustellen App erhält. Die Mobile Applikation wird mit dem Framework .NET Maui von Microsoft realisiert. Dies ermöglicht den Bau einer mobilen Applikation für iOS und Android aus einer Codebase.

#### Auslieferungsstrategie

Das Backend wird in einer public Cloud auf einer Container Plattform betrieben. Das Webfrontend ist über die gleiche Domäne wie das Backend verfügbar und kann von jedem Rechner mit Verbindung zum Internet aufgerufen werden. Die mobile App wird als Download im Cloudstorage zur Verfügung gestellt. Mitarbeiter können die App über einen offiziellen Link auf ihr Gerät herunterladen und installieren.

### Bausteinsicht

### Architekturentscheidungen

Im folgenden Abschnitt werden die wichtigsten Architekturentscheidungen für die Umsetzung der Baustellen App dokumentiert.

#### Aufteilung des Backends in «Micro-Services»

Tabelle Entwurfsentscheidung: Aufbau des Backends als Micro Service Architecture

|  |  |
| --- | --- |
| ID | ADR-002 |
| Titel | Aufbau des Backends als Micro Service Architecture (MSA) |
| Kontext | Für den Aufbau des Backends stehen folgende Architekturen zur Auswahl:   1. **Micro Service Architecture**   Eine MSA hat einige wichtige Charakteristiken. MSA gilt als Wart- und Erweiterbar weil die Fachliche Logik der Anwendung auf voneinander unabhängige Services verteilt wird [12]. Insbesondere für komplexere Systeme eignet sich eine MSA [13] und bietet auf längere Sicht sollen auch die Kosten für eine MSA geringer ausfallen [13].   1. **Monolith**   Eine monolithische Architektur zeichnet sich dadurch aus, dass die Funktionen der Anwendung in einem ausführbaren Software System gekapselt sind und dieses System aus einer gemeinsamen Codebasis besteht [13]. Für ein simples Software System bietet sich diese Architektur an, da die Time-To-Market für ein einfaches Software System generell kurz ist [13].   1. **Serverless Architecture**   Eine Serverlose Architektur macht sich das PaaS [14] Angebot von Cloud Anbietern zu nutze. Eine Serverless Architecture bietet eine ser hohe Skalierbarkeit. Die Verantwortung über die Verfügbarkeit und Sicherheit der Serivces wird an den Platformbetreiber übergeben [14]. |
| Entscheidung | Das Backend der Baustellen App wird als Micro Service Architektur umgesetzt. |
| Begründung | Die Baustellen App wird in einem ersten Schritt als MVP entwickelt und soll in Zukunft weiter ausgebaut werden. Erweiterbarkeit ist deshalb von Zentraler bedeutung. MSA bietet eine gute Grundlage um eine verfügbare, performante und erweiterbare Applikation. |
| Konsequenz | Initial ist der Aufwand für die Umsetzung einer MSA höher, weil der zusätzliche Aufwand, insbesondere im Bereich Kommunikation unter den Services, hoch ist. Der Aufwand zahlt sich auf lange Sicht aus. |
| Alternativen | Monolith |

#### Einsatz von «asynchroner Datensynchronisation»

Wie im Kapitel 2.2.1 beschrieben steht für die Datensynchronisation zwei Muster zur Verfügung. In diesem Abschnitt ist der Entscheid für das asynchrone Entwurfsmuster beschrieben.

Tabelle Entwurfsentscheidung: Einsatz von Asynchroner Datensynchronisation

|  |  |
| --- | --- |
| ID | ADR-002 |
| Titel | Verwendung von Asynchroner Datensynchronisation |
| Kontext | Zach McCormick und Douglas C. Schmidt beschreiben zwei Entwurfsmuster für die Datensynchronisation [5].   1. **Synchrone Datenübertragung**   Die synchrone Datenübertragung ist im Kapitel 2.2.1.1 beschrieben. Anwendung wartet nach der Initialisierung von Datensynchronisation auf die Antwort. Wärend des Synchronisationsvorgangs ist die Benutzerinteraktion blockiert.   1. **Asynchrone Datenübertragung**   Die asynchrone Datenübertragung ist im Kapitel 2.2.1.2 beschrieben. Die Synchronisation von Daten wird in eine Subprozess initiiert, wärend die Benutzeroberfläche für den Benutzenden interagierbar bliebt. |
| Entscheidung | Für die Datensynchronisation der Baustellen App wird die asynchrone Datensynchronisation verwendet. |
| Begründung | Die Helion AG setzt eine bedienerfreundliche Benutzeroberfläche voraus wie in den Qualitätszielen beschrieben (QZ-003). Deshalb wird auf das asynchrone Datensychnronisationspattern in der Umsetzung der Baustellen App gesetzt. |
| Konsequenz | Das Statemanagement der Applikation wird etwas komplexer. Sollte es sich zeigen, das gewisse Module der Baustellen App auf komplette Datensätze oder dringend auf die Antwort einer Schnittstelle angewiesen sind, muss das im Statemanagement berücksichtig werden. |
| Alternativen | Synchrone Datenübertragung |

#### Einsatz von «Complete Storage» für den mobilen Client

Tabelle Entwurfsentscheidung: Complete Storage für den mobilen Client

|  |  |
| --- | --- |
| ID | ADR-002 |
| Titel | Anwendung des Complete Storage Pattern |
| Kontext | Zach McCormick und Douglas C. Schmidt beschreiben zwei Entwurfsmuster für die Datenspeicherung [5].   1. **Partial Storage**   Das Entwurfsmuster Partial Storage ist im Kapitel 2.2.2.1 beschrieben. Prinzipiel werden Daten nur geladen wenn sie benötigt werden. Dadurch wird der benötigte Speicher optimiert.   1. **Complete Storage**   Das Entwurfsmuster Complete Storage ist im Kapitel 2.2.2.2 beschrieben. Dies erhöht die Verfügbarkeit der Daten auch wenn die Netzwerk Konnektivität unterbrochen ist. |
| Entscheidung | Für den mobilen Client wird das Entwurfsmuster Complete Storage angewendet. |
| Begründung | Die wichtigste Anforderung an die mobile App ist die Offline Fähigkeit (QZ-001). Ein von Helion AG bestätigter Fakt ist, dass die Konnektivität der Geräte auf der Baustelle oft unterbrochen ist. Deshalb kommt für den mobilen Client nur das Muster Complete Storage in Frage, da eine Abfrage der Daten nach Bedarf oft nicht möglich ist. |
| Konsequenz | Der Speicherbedarf auf den mobilen Endgeräten ist begrenzt. Das muss in der Umsetzung berücksichtigt werden. Entsprechend muss fehlender Speicherplatz in der mobilen App behandelt werden. |
| Alternativen | Anzahl der zu speichernden Datensets beschränken. |

#### Einsatz von «Timestamp Transfair» für den mobilen Client

Der mobile Client von der «Baustellen App» hat die bedeutende Eigenschaft, dass der Client für unbestimmte Zeit offline sein kann. Wärend dieser Zeit können Datensätze auf dem mobilen Client und im Backend auseinander divergieren. Deshalb ist die art der Datenübertragung und Zusammenführung zwischen dem mobilen Client und dem Backend von grosser Bedeutung.

Als Basis für den Datentransfer wurden drei Verfahren in Betracht gezogen:

Tabelle Varianten für die Datenübertragung zwischen mobile Client und Backend

|  |  |  |
| --- | --- | --- |
| ID | Variante | Beschreibung |
| V1 | «Full Transfer» | Siehe Kapitel 2.2.3.1 |
| V2 | «Timestamp Transfer» | Siehe Kapitel 2.2.3.2 |
| V3 | «Mathematical Transfer» | Siehe Kapitel 2.2.3.3 |

Um diese Varianten objektiv zu vergleichen und zu bewerten, wurde eine Nutzwertanalyse durchgeführt. In der folgenden Tabelle werden die relevanten Kriterien für die Auswahl aufgezeigt.

Tabelle Bewertungskriterien bei der Auswahl des Datentransfermusters

|  |  |  |
| --- | --- | --- |
| ID | Bewertungskriterium | Beschreibung |
| K1 | Komplexität | Die betrachteten Lösungsvarianten unterscheiden sich in Bezug auf ihre Komplexität. Das Bewertungskriterium ist aus folgenden Gründen bedeutsam:   1. **Entwicklungszeit und Kosten**   Die Entwicklung komplexer Software erfordert oft viel Zeit und Ressourcen. Es wird berücksichtigt, wie sich die Wahl der Varainte auf die Entwicklungskosten und den Aufwand auswirken würde.   1. **Wartbarkeit**   Je komplexer eine Software ist, umso schwieriger kann es sein, Fehler zu finden und Anpassungen vorzunehmen. Je komplexer eine Software ist umso aufwendiger ist es die Software zu Warten.   1. **Fehleranfälligkeit**   Eine komplexe Software enthält tendenziell mehr Punkte an denen Fehler entstehen können. Einfache Architekturen sind oft robuster und weniger fehleranfällig.  **Bewertungsskala:**  1 = hohe Komplexität  2 = mittlere Komplexität  3 = geringe Komplexität |
| K2 | Speicherbedarf | Die Betrachteten Lösungsvarianten unterscheiden sich in Bezug auf den zusätzlichen Speicherbedarf. Dieses Kriterium ist aus folgenden Gründen wichtig:   1. **Zusätzlicher Speicher**   Gewisse Algorithmen benötigen für ihre Funktion zusätzlichen Speicherplatz. Der benötigte Speicherplatz kann sich mit dem Typ des Datensatzes stark verändern.   1. **Arbeitsspeicher (RAM)**   Kann der Algorithmus seine Daten im Arbeitsspeicher halten wärend der Verarbeitung oder verlangt die Menge an Daten eine Auslagerung in eine Datenbank. Dies hat Einfluss auf die Effizienz des Algorithmus.  **Bewertungsskala:**  1 = hoher zusätzlicher Speicherbedarf (RAM und Datenbank)  2 = mittlerer zusätzlicher Speicherbedarf (RAM oder Datenbank)  3 = kein zusätzlicher Speicherbedarf |
| K3 | Rechenleistung | Die betrachteten Lösungsvarianten unterscheiden sich in Bezug auf die Rechenleistung. Aus folgenden Gründen ist dieses Kriterium von Bedeutung:   1. **Auswirkung auf die Verfügbarkeit**   Dauert die Berechnung für die Zusammenführung lange, hat dies Auswirkungen auf die Antwortzeiten.   1. **Einfluss von grosser Divergenz**   Die Zusammenführung von Datensets kann mit zunehmender Divergenz Rechenintensiver werden.   1. **Einfluss auf die Skalierung**   Besonders bei wiedererlangen der Konnektivität wird der Datentransfer und damit das Zusammenführen von Datensätzen erwartet. Dies kann zu einem Performance Peak führen.  **Bewertungsskala:**  1 = hohe Rechenleistung  2 = mittlere Rechenleistung  3 = geringe Rechenleistung |
| K4 | Netzwerkbandbreite | Die betrachteten Lösungsvarianten unterscheiden sich in Bezug auf die benötigte Netzwerkbandbreite. Aus folgenden Gründen ist dieses Kriterium von Bedeutung:   1. **Kosten**   Für mobile Geräte entstehen für die Benutzung von Netzwerkbandbreite oft Kosten.   1. **Verfügbare Bandbreite**   Die tatsächlich verwendbare Bandbreite in einem mobilen Netz kann stark variieren.  **Bewertungskriterien:**  1 = hoher Bandbreitenbedarf  2 = mittlerer Bandbreitenbedarf  3 = geringer Bandbreitenbedarf |
| K5 | Geeignetes Konfliktmanagement | Die betrachteten Lösungsvarianten unterscheiden sich im Konfliktmanagement und in dessen Eignung für den mobile Client. Das Kriterium wird aus den folgenden Gründen bewertet:   1. **Konkurierende Änderungen**   Konkurierende Änderungen sind wahrscheinlich im Anwendungsszenario der Baustellen App. Die Datenübertragung muss mit konkurierende Änderungen umgehen können.   1. **Potenzieller Datenverlust**   Änderungen am selben Datensatz von unterschiedlichen Quellen kann zum Verlust von Daten führen. Die akzeptanz und der Umfang von Datenvelust muss mit den Anforderungen der Helion AG überinstimmen.  **Bewertungskriterien:**  1 = ungeeignetes Konfliktmanagement  2 = eingeschränkt geeignetes Konfliktmanagement  3 = geeignetes Konfilkmanagement |

In den folgenden Tabellen wird die Bewertung der Varianten im Bezug auf die oben beschriebenen Kriterien aufgezeigt.

Tabelle Bewertung der Komplexität der Varianten

|  |  |  |
| --- | --- | --- |
| Variante | Komplexität | Begründung |
| «Full Transfer» | Geringe Komplexität | Wie in [5] gezeigt verwendet das Muster «Full Transfer» keine Algorithmen für die Bestimmung der zusammen zu führenden Daten. Entsprechend ist die Implementierung einfach und die Fehleranfälligkeit gering. |
| «Timestamp Transfer» | Mittlere Komplexität | Für die Implementierung wird ein Algorithmus benötigt, welcher Zeitstempel vergleicht [5]. Ausserdem werden Zusätzliche Endpoints für den Austausch der Zeitstempel benötigt. |
| «Mathematical Transfer» | Hohe Komplexität | Diese Variante verwendet komplexe mathematische Algorithmen [2] für die Berechnung der Zusammenführung von Datensätzen. Dies führt zu hohem Entwicklungsaufwand und insgesamt zu einem Komplexeren System [5]. |

Tabelle Bewertung des Speicherbedarfs der Varianten

|  |  |  |
| --- | --- | --- |
| Variante | Speicherbedarf | Begründung |
| «Full Transfer» | Kein zusätzlicher Speicherbedarf | Diese Variante muss keine zusätzlichen Daten für die Datenübertragung abspeichern, weil der Datensatz jeweils als ganzes übermittelt wird [5]. |
| «Timestamp Transfer» | Mittlerer zusätzlicher Speicherbedarf | Damit der Zeitstempel verglichen werden kann, muss jeder Datensatz ein «Erstellt am» und «Modifiziert am» Datumsfeld enthalten [5]. Dadurch ist ein zusätzlicher Speicherbedarf pro Datensatz nötig. |
| «Mathematical Transfer» | Hoher zusätzlicher Speicherbedarf | Die Berechnung der Zusammenführung der Datensätzen benötigt oft zusätzlichen Speicher oder sogar separate Tabellen in der Datenbank [8]. Der Bedarf an Speicher kann mit der Zuhname der Differenz oder der Menge an Datensätze steigen [8]. |

Tabelle Bewertung der Performance der Varianten

|  |  |  |
| --- | --- | --- |
| Variante | Rechenleistung | Begründung |
| «Full Transfer» | Geringe Rechenleistung | Es findet wird keine Rechenleistung für die Ausführung einer Zusammenführung benötigt [5]. |
| «Timestamp Transfer» | Geringe Rechenleistung | Der Vergleich der Timestamps benötigt keinen nennenswerten Mehraufwand von Rechenleistung [5]. |
| «Mathematical Transfer» | Hohe Rechenleistung | Die Berechnung für die Zusammenfühung kann je nach Algorithmus höher ausfallen oder mit der Zuhname von Datensätzen zunehmen [15]. |

Tabelle Bewertung der Netzwerkbandbreite der Varianten

|  |  |  |
| --- | --- | --- |
| Variante | Netzwerkbandbreite | Begründung |
| «Full Transfer» | Hoher Bandbreitenbedarf | Gemäss [5] hat das «Full Transfer» Muster einen hohen Bedarf an Netzwerkbandbreite. Dies beruht auf der Tatsache, dass die Datensätze in ihrer Gänze übertragen werden. |
| «Timestamp Transfer» | Mittlerer Bandbreitenbedarf | Gemäss [5] benötigt dieses Muster weniger Banbreite als «Full Transfer». In diesem Muster werden nur die Datensätze übertragen, welche sich verändert haben oder neu dazu gekommen sind [5]. |
| «Mathematical Transfer» | Geringer Bandbreitenbedarf | Dieses Muster hat das potenzial den geringten Bandbreitenbedarf aufzuweisen [5]. Durch mathematische Algorithmen kann die Entscheidung, welche Daten übertragen werden, sehr granular ausfallen [5]. |

Tabelle Bewertung der Eignung des Konfliktmanagements der Varianten

|  |  |  |
| --- | --- | --- |
| Variante | Konfliktmanagent | Begründung |
| «Full Transfer» | Ungeeignet | Der «Full Transfer» überschreibt vorhandene Datensätze im Moment der übertragung [5]. Im Fall der Baustellen App würde dies zu unvorhersehbarem Datenverlust führen. Da mit dieser Variante zum Zeitpunkt der Synchronisierung nicht bestimmt werden kann, welche Änderung, die letzte aktuell gültige ist. |
| «Timestamp Transfer» | Geeignet | Mit dem «Timestamp Transfer» wird bei einem Konflikt der zuletzt geänderte Datensatz angewendet («Last Write Wins») [5]. Aus den Anforderungen der Helion AG kann dieses Verhalten als das Gewünschte verhalten bezeichnet werden, da keine Interaktion durch die Anwendenden nötig wäre. |
| «Mathematical Transfer» | Eingeschränkt Geeignet | «Mathematical Transfer» hat das potenzial Konflikte sehr granular aufzulösen [5]. Kokurierende Änderungen sind auch für mathematische Algorithmen nicht immer zu 100% auflösbar und erfordern entweder eine Regel oder manuelles Eingreifen zur Auflösung. |

In der folgenden Tabelle wird die gewählte Gewichtung der Kriterien aufelistet und Begründet.

Tabelle Aulistung und Begründung der Gewichtung der Kriterien

|  |  |  |
| --- | --- | --- |
| Kriterium | Gewichtung | Begründung |
| Komplexität | 20% | Da sich die Komplexität auf Aufwand und Kosten der Implementierung auwirkt und die zukünftige Erweiterbarkeit, ist dieses Kriterium hat dieses Kriterium eine höhere Gewichtung. |
| Speicherbedarf | 5% | Der Speicherbedarf wirkt sich im Endeffekt auf die Betriebskosten der Plattform aus. Deshalb wird dieses Kriterium betrachtet, hat jedoch eine geringe Gewichtung. |
| Rechenleistung | 5% | Die benötigte Rechenleistung wirkt sich im Endeffekt auf die Betriebskosten der Plattfrom aus. Deshalb wird dieses Kriterium betrachtet, hat jedoch eine geringe Gewichtung. |
| Bandbreite | 20% | Die verfügbare Bandbreite ist eine gegebene Grösse. Deshalb ist die effiziente Nutzung der verfügbaren Bandbreite eine wichtige Voraussetzung und wird entsprechend höher Gewichtet. |
| Konflikmanagement | 50% | Das Konfliktmanagement ist für die Helion AG ein zentraler Punkt. Benutzerinteraktion sollte möglichst keine erforderlich sein. Trotzdem muss die unumgängliche Divergenz zwischen Datensätzen gehandhabt werden. Deshalb ist dieses Kriterium sehr hoch gewichtet. |

Basierend auf der Bewertung der Kriterien ergibt sich folgendes Gesamtergebnis:

Tabelle Ergebnis der Nutzwertanalyse

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Kirterium | Gewicht | «Full Transfer» | | «Timestamp Transfer» | | «Mathematical Transfer» | |
|  |  | Punkte | Gewichtet | Punkte | Gewichtet | Punkte | Gewichtet |
| Komplexität | 20% | 3 | 0.6 | 2 | 0.4 | 1 | 0.2 |
| Speicherbedarf | 5% | 3 | 0.15 | 2 | 0.1 | 1 | 0.05 |
| Rechenleistung | 5% | 3 | 0.15 | 3 | 0.15 | 1 | 0.05 |
| Bandbreite | 20% | 1 | 0.2 | 2 | 0.4 | 3 | 0.6 |
| Konfliktmanagement | 50% | 1 | 0.5 | 3 | 1.5 | 2 | 1 |
| **Summe** | **100 %** | **11** | **1.6** | **12** | **2.55** | **8** | **1.9** |

Die Nutzwertanalyse zeigt, dass die Variante «Timestamp Transfer» für die Implementierung in die «Baustellen App» am besten geeignet ist. Aus dieser Analyse ergibt sich folgende Architekturentscheidung:

Tabelle Entwurfsentscheidung: Einsatz von Timestamp Transfer für den Datentransfer

|  |  |
| --- | --- |
| ID | ADR-004 |
| Titel | Anwendung des «Timestamp Transfer» Entwurfsmuster |
| Kontext | Der Datentransfer ist von zentraler Bedeutung für die Datensynchronisation zwischen der mobilen App und dem Backend der «Baustellen App». |
| Entscheidung | Für den Datentransfer wird das Entwurfsmuster «Timestamp Transfer» angewendet. |
| Begründung | Die Wahl des Datentransfer Entwurfsmuster wurde auf Grund der oben aufgeführten Nutzewertanalyse getroffen. |
| Konsequenz | Der Datentransfer erfolgt gemäss dem «Timestamp Transfer» Entwurfsmustere. Für die Zusammenführung von Datensätzen wird die «Last Write Wins» Regel angewendet. |
| Alternativen | Folgende Varianten wurden betrachtet:   * «Full Transfer» * «Timestamp Transfer» * «Mathematical Transfer» |

### Risiken

# Diskussion / Zusammenfassung

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit.

## Überschrift

Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit.

Quatur ad quibusamus, et exerionem eostis peror sedipis aut int la peris eatibusam is aut autem imporum soluptatium coritas perepratem doluptas sitatur atium, ilitat velenihictem eaquas molor serit doloratiis abo.

## Überschrift

Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit.

# Abbildungsverzeichnis

# Tabellenverzeichnis

[Tabelle 1 Anforderungen an die Authentifizierung 9](#_Toc188879484)

[Tabelle 2 Anforderungen an die Projektübersicht 10](#_Toc188879485)

[Tabelle 3 Anforderungen an die Projektdetails 10](#_Toc188879486)

[Tabelle 4 Anforderung zu den Notizen 10](#_Toc188879487)

# Glossar

**Backend**

Als Backend wird die Softwarekomponente eines verteilten Systems bezeichnet, welche auf einem Server betrieben wird und über ein privates Netzwerk oder das Internet erreichbar ist. Üblicherweise ist das Backend für die Persistierung von Daten zuständig und enthält unter Umständen Geschäftslogik, welche die ankommenden oder ausgehenden Daten entsprechend verarbeitet.

**Desing Patter**  
Auf Deutsch Entwurfsmuster, ist ein Begriff aus der Softwareentwicklung. Ein Entwurfsmuster ist wie der Name sagt ein Muster um ein bestimmtes Problem zu lösen. Das Design Pattern beschreibt die Lösung ohne konkrete Implementierungen. Dadurch kann das Entwurfsmuster unabhängig von der Plattform eingesetzt werden.

**Frontend**

Das Frontend ist der Teil eines Softwaresystems, welches für die Interaktion mit den Anwendern zuständig ist. Es stellt eine Benutzeroberfläche zur Verfügung.

**MVP**

Minimum Viable Product ist die englische Beschreibung eines Produkts mit dem minimal nötigen Funktionsumfang, um von Kunden oder Benutzern verwendet zu werden und Rückmeldung für die zukünftige Funktionsentwicklung zu gewinnen. Dieser Term wird üblicherweise in der Software Entwicklung verwendet.

**SAD**

Das Software Architektur Dokument ist das Ergebnis der Arbeit des Software Architekten. Das Dokument gibt darüber Auskunft, wie die Qualitätsanforderungen an die Applikation umgesetzt werden sollen. Es dokumentiert die relevanten Architekturentscheidungen, analysiert Risiken, definiert Massnahmen zur Risikominderung und enthält weitere Querschnittsthemen, welche aus Sicht des Architekten für die Applikation von Bedeutung sind.

**UI/UX**

UI ist die Abkürzung für User Interface. Zu Deutsch Benutzeroberfläche. UX steht für User Experience und beschreibt die Benutzererfahrung. Beide Begriffe zusammen behandeln das Aussehen und Verhalten von grafischen Oberflächen von Softwaresystemen, welche von menschlichen Anwendern verwendet werden.

**Use Cases**

Zu Deutsch Anwendungsfälle. Ein Anwendungsfall beschreibt eine Funktionalität eines Systems, welcher Aktor oder welches System diese Funktionalität verwendet

# Literaturverzeichnis

[1] «Data Syncing in Core Data Based iOS Apps | DENIVIP Media». Zugegriffen: 19. Oktober 2024. [Online]. Verfügbar unter: http://blog.denivip.ru/index.php/2014/04/data-syncing-in-core-data-based-ios-apps/?lang=en

[2] A. Selvan, «Set reconciliation in multi-node environment», in *2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, Juli 2013, S. 1–4. doi: 10.1109/ICCCNT.2013.6726730.

[3] S. Gilbert und N. Lynch, «Perspectives on the CAP Theorem», *Computer*, Bd. 45, Nr. 2, S. 30–36, Feb. 2012, doi: 10.1109/MC.2011.389.

[4] «Was ist das CAP-Theorem? | IBM». Zugegriffen: 25. Januar 2025. [Online]. Verfügbar unter: https://www.ibm.com/de-de/topics/cap-theorem

[5] Z. McCormick, «Data Synchronization Patterns in Mobile Application Design».

[6] D. Hasanenko, «Data Synchronization Patterns», Medium. Zugegriffen: 1. Juli 2024. [Online]. Verfügbar unter: https://hasanenko.medium.com/data-synchronization-patterns-c222bd749f99

[7] E. Freeman *u. a.*, *Head first design patterns: building extensible & maintainable object-oriented software*, Second edition. Sebastopol, CA: O’Reilly Media, 2020.

[8] M. Mitzenmacher und R. Pagh, «Simple multi-party set reconciliation», *Distrib. Comput.*, Bd. 31, Nr. 6, S. 441–453, Nov. 2018, doi: 10.1007/s00446-017-0316-0.

[9] «Kanban», *Wikipedia*. 12. Januar 2025. Zugegriffen: 27. Januar 2025. [Online]. Verfügbar unter: https://de.wikipedia.org/w/index.php?title=Kanban&oldid=252173638

[10] «Use case diagrams are UML diagrams describing units of useful functionality (use cases) performed by a system in collaboration with external users (actors).» Zugegriffen: 31. Dezember 2024. [Online]. Verfügbar unter: https://www.uml-diagrams.org/use-case-diagrams.html

[11] D. G. Starke, «arc42 Template Overview», arc42. Zugegriffen: 27. Januar 2025. [Online]. Verfügbar unter: https://arc42.org/overview

[12] A. Razzaq und S. A. K. Ghayyur, «A systematic mapping study: The new age of software architecture from monolithic to microservice architecture—awareness and challenges», *Comput. Appl. Eng. Educ.*, Bd. 31, Nr. 2, S. 421–451, 2023, doi: 10.1002/cae.22586.

[13] «Monolithisch vs. Microservices - Unterschied zwischen Softwareentwicklungsarchitekturen - AWS», Amazon Web Services, Inc. Zugegriffen: 29. Januar 2025. [Online]. Verfügbar unter: https://aws.amazon.com/de/compare/the-difference-between-monolithic-and-microservices-architecture/

[14] «Was ist Serverless Computing? – Serverless Computing erklärt – AWS», Amazon Web Services, Inc. Zugegriffen: 29. Januar 2025. [Online]. Verfügbar unter: https://aws.amazon.com/de/what-is/serverless-computing/

[15] M. Mitzenmacher und G. Varghese, «The complexity of object reconciliation, and open problems related to set difference and coding», in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Okt. 2012, S. 1126–1132. doi: 10.1109/Allerton.2012.6483345.

# Anhang

## Sourcecode

Der Sourcecode zum PoC der Baustellen App befindet sich in folgendem Git Repository

<https://github.com/zuercheram/bauestellen-app>

# Verwendung von KI

In dieser Arbeit wurde auf die Verwendung von KI verzichtet.

# Selbständigkeitserklärung

Ich bestätige, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der im Literaturverzeichnis angegebenen Quellen und Hilfsmittel angefertigt habe. Sämtliche Textstellen, die nicht von mir stammen, sind als Zitate gekennzeichnet und mit dem genauen Hinweis auf ihre Herkunft versehen.

Ich bestätige weiterhin, dass ich bei der Erstellung dieser Studienarbeit durchgehend steuernd gearbeitet habe und von einer KI erzeugte Texte bzw. Textfragmente nicht unreflektiert übernommen habe.

Ort, Datum:

1. https://www.figma.com/ [↑](#footnote-ref-1)