



Health Ledger

Die Digitale Patientenakte.

Cem Basoglu, Mario Cichonczyk, Matthias Kersting, Nils Kirchhof, Fynn Klöpper, Kevin Schima, Patrick Starzynski

Problemstellung - IST-Zustand

1. Krankenakte ist tief in die persönliche Privatsphäre eingebettet.
2. Kontrolle über Krankenakte liegt nicht beim Patienten. Daten sind undurchsichtig verteilt.
3. Zentrale Lösungen sind keine Antwort auf das Problem. Die unbekannte Motivation der Betreiber und deren Einflussnahme erzeugt Misstrauen
4. Medizinische Informationsüberlassungen wären lediglich ein gewährter Dienst einer Körperschaft anstatt sich in echter Kontrolle durch den Informationseigner zu konkretisieren.

Zielsetzung - SOLL-Zustand

1. Die unkontrollierbare, nicht nachvollziehbar verteilte Patienteninformation wird durch Digitalisierung in eine elektronische, konzeptionell zentrale Akte überführt.
2. Kontrolle über die Freigabe der Daten liegt mit variabel wählbarer Transparenzgranularität beim Informationseigner.
3. Global einsehbar für zügige und risikofreie Diagnosestellung.
4. Smarter Vernetzung aller Beteiligten für beschleunigte Genesung.
5. Durch Chaincode-Verschreibungen wird die papierlose Medikation ermöglicht. Nicht nur ist die Wahrscheinlichkeit einer Falschverabreichung rigoros eingedämmt, der Verarbeitungskomfort für Apotheker und Versicherer steigt zusätzlich deutlich.
6. Maßgeschneiderte Oberflächen für das Anwenderspektrum zwischen medizinischem Laien, Facharzt und Versicherungsdienstleister.
7. Kurzeinsicht für Krankschreibungen.

Health Ledger

Die Digitale Patientenakte.

Konzeptionelle
Herausforderungen

Transparenzgranularität

1. Steuerung über Businesslogik

Patient definiert welcher Teilnehmer, welche Informationen der Patientenakte einsehen darf und die Businesslogik (Chaincode) stellt diese Funktionalität sicher.

⇒ Da jeder Peer-Betreiber eine Kopie des Distributed Ledgers vorliegen hat, kann die Businesslogik umgangen werden

Transparenzgranularität

2. Verschlüsselte Patientenakte

Die Patientenakte wird verschlüsselt im Distributed Ledger abgelegt und nur die berechtigten Teilnehmer erhalten den Schlüssel um auf Informationen zuzugreifen.

⇒ Mehrere Schlüssel nötig um Abstufungen in der Transparenz zu ermöglichen und einmal gewährte Einsicht kann nicht mehr entzogen werden.

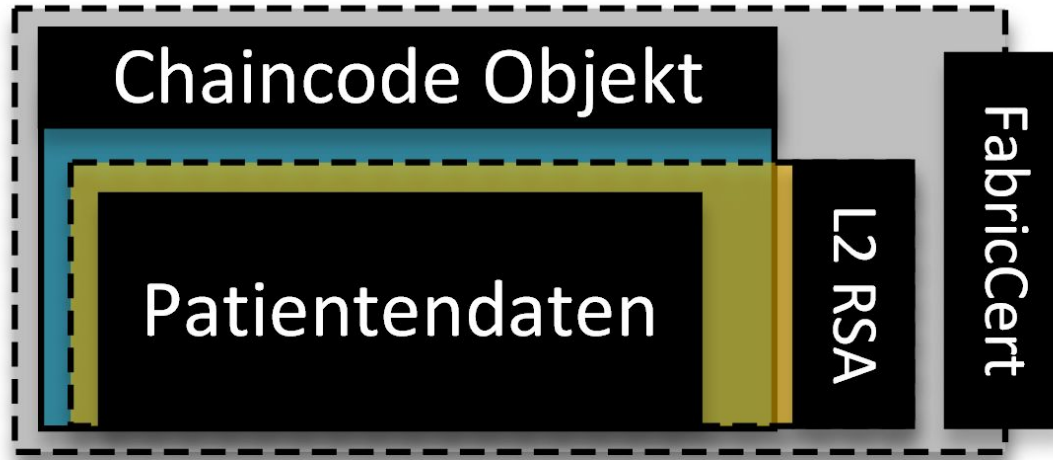
Transparenzgranularität

3. Verschlüsselte Patientenakte und Einsichtsanhfragen

Die Patientenakte wird verschlüsselt im Distributed Ledger abgelegt und nur der Patient hält den Schlüssel um auf Informationen zuzugreifen. Bei Einsichtsanhfragen durch andere Teilnehmer werden die vom Patienten freigegeben Daten mit dem öffentlichen Schlüssel des anfragenden Teilnehmers verschlüsselt.

⇒ Es wird immer nur die Einsicht auf einen Snapshot der Patientenakte für einen bestimmten Teilnehmer gewährt. Änderungen an der Patientenakte müssen erneut angefragt und freigegeben werden.

Transparenzgranularität

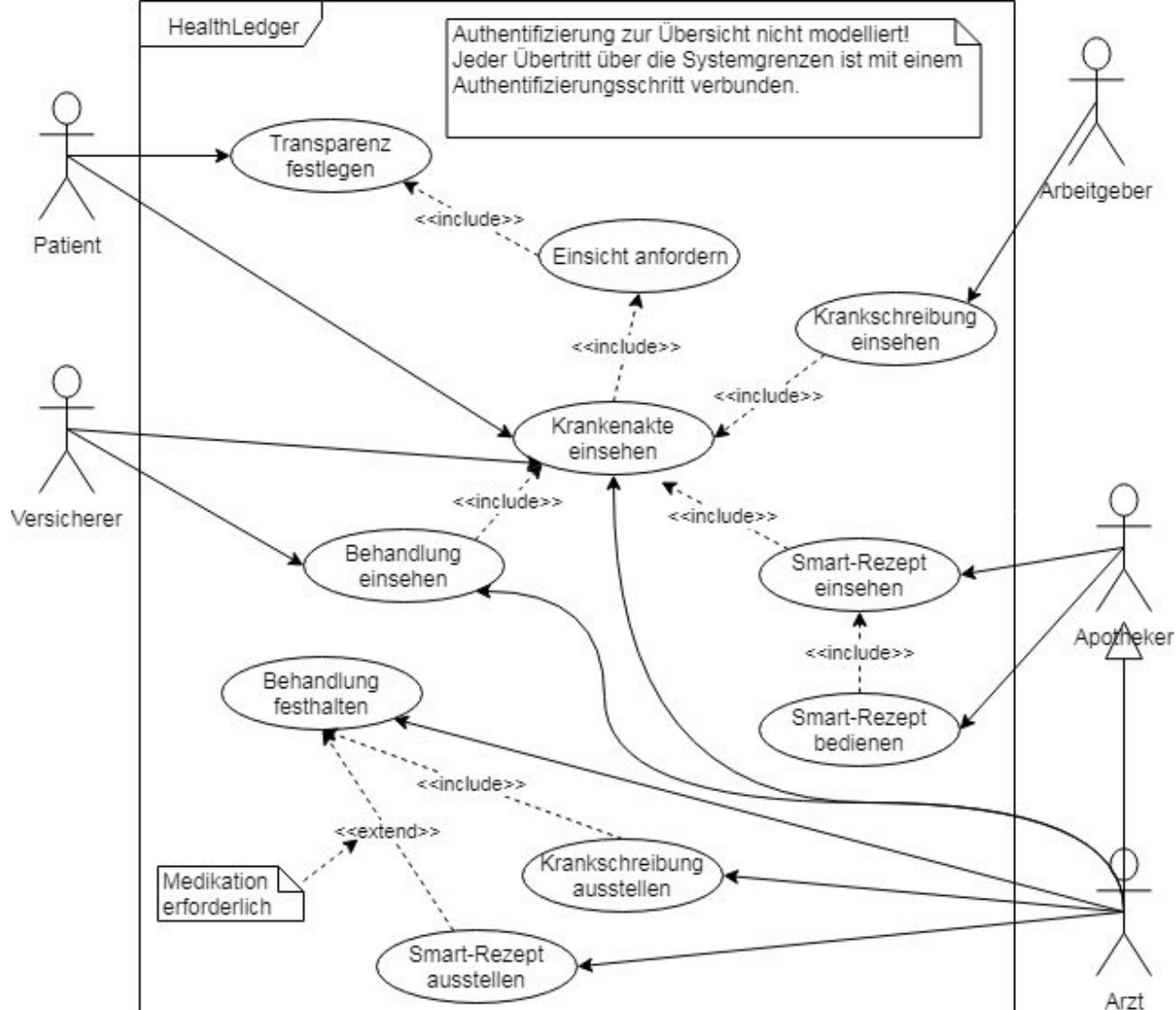


Health Ledger

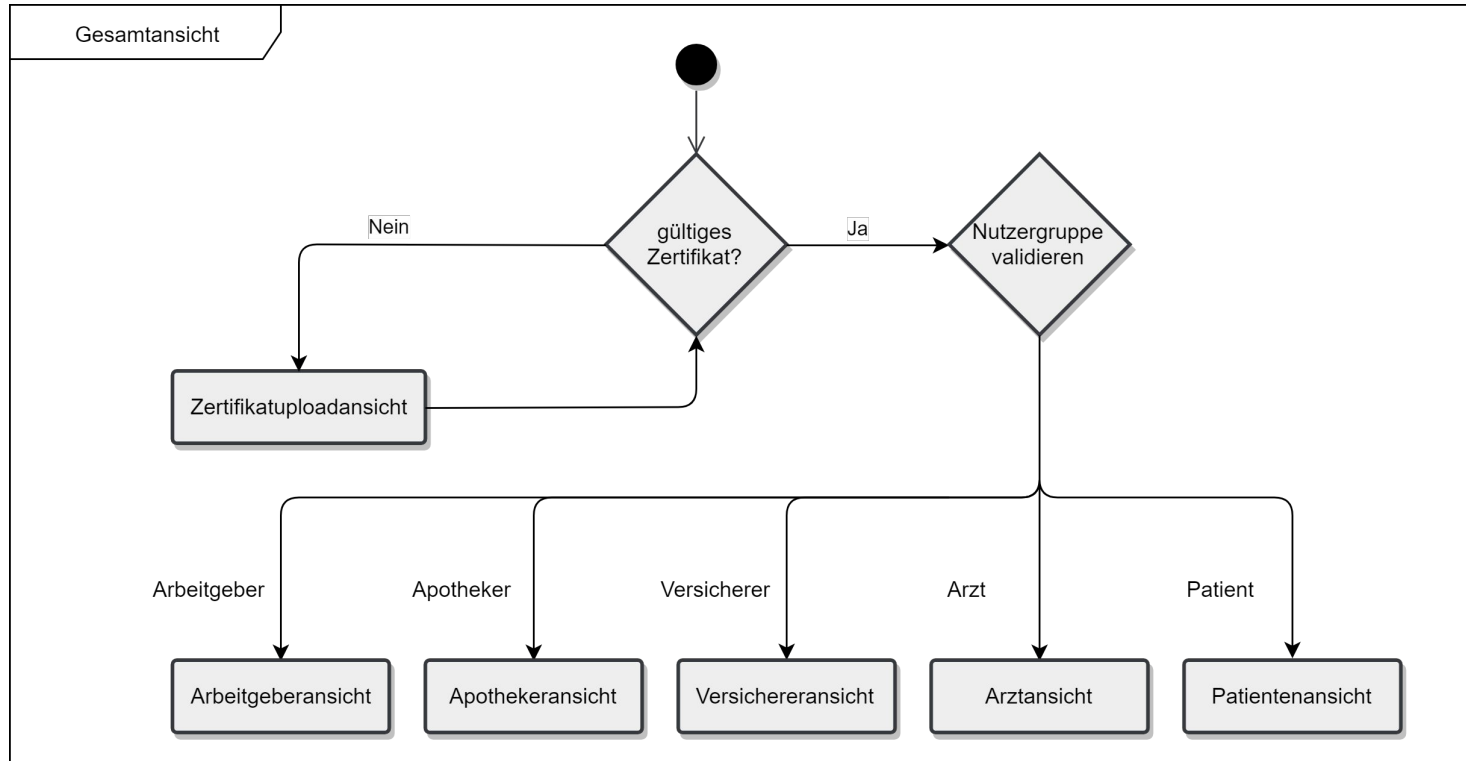
Die Digitale Patientenakte.

Technische
Umsetzung

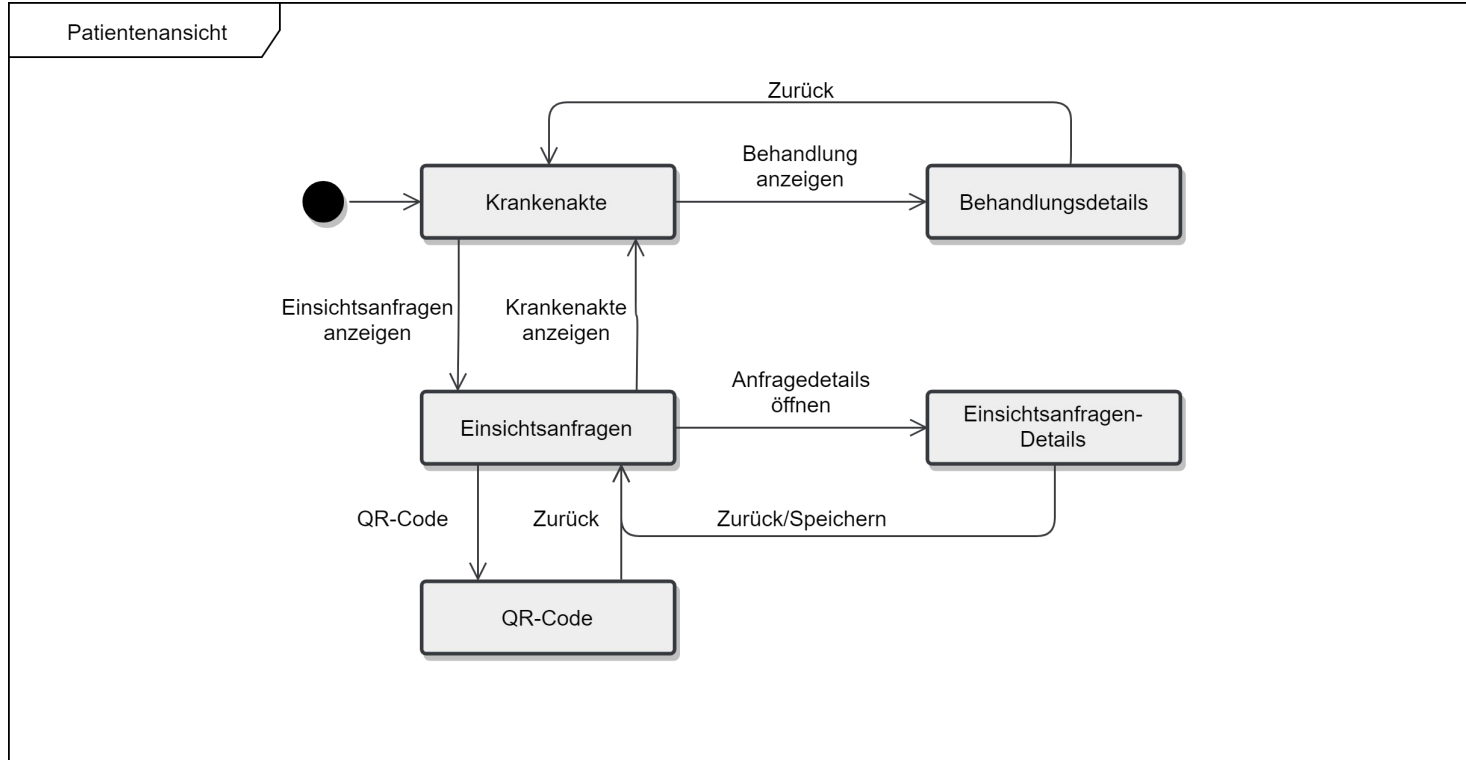
UseCases



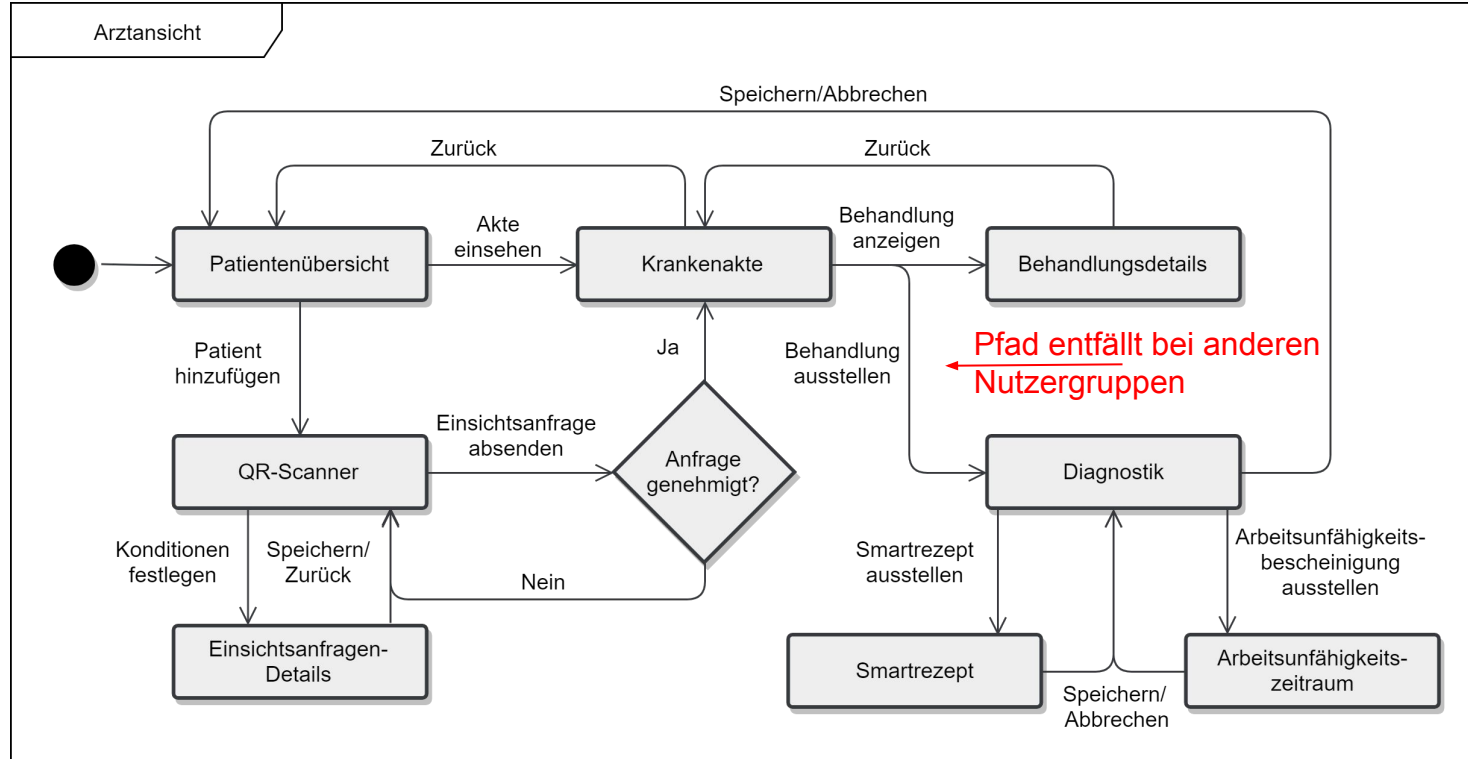
Zustandsdiagramm - Gesamtansicht



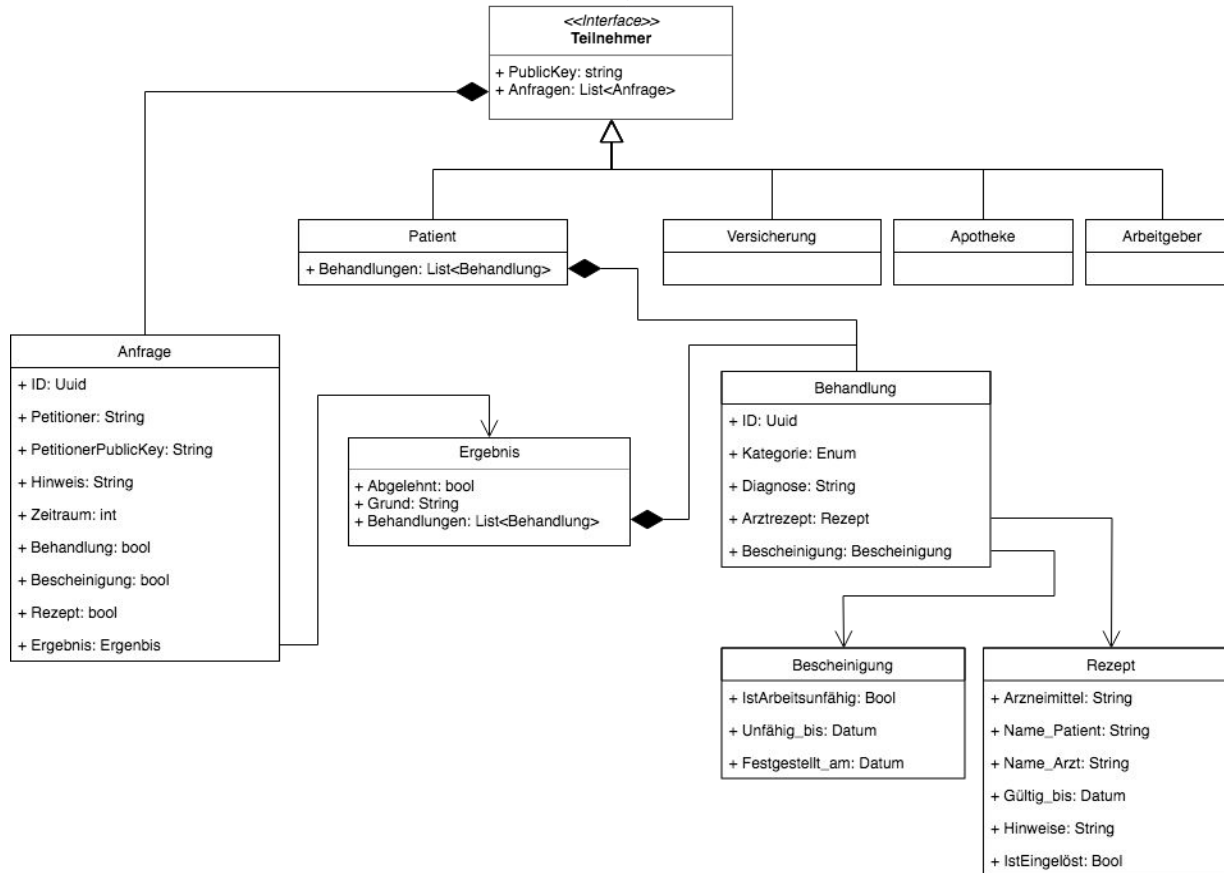
Zustandsdiagramm - Patientenansicht



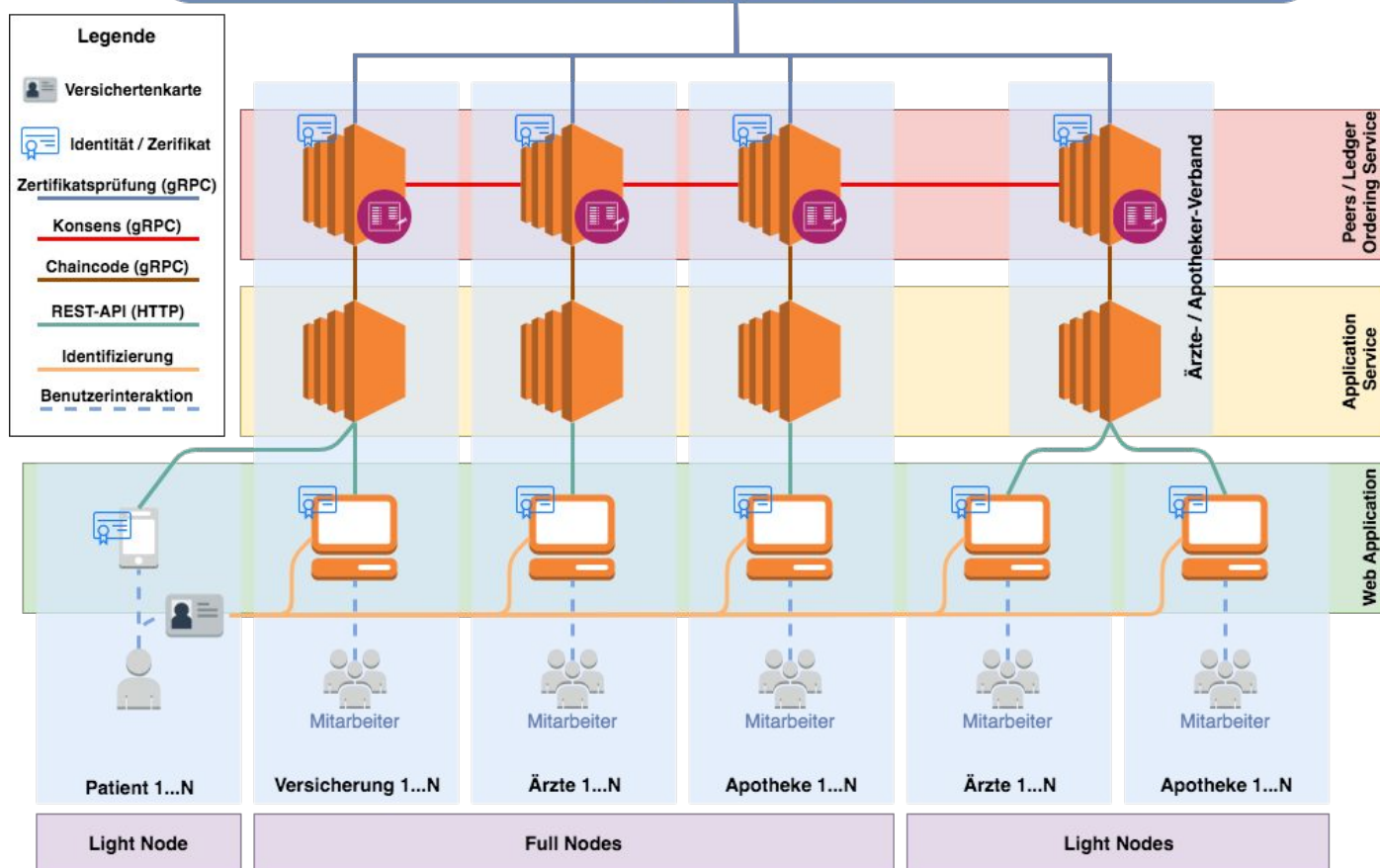
Zustandsdiagramm - Arztansicht



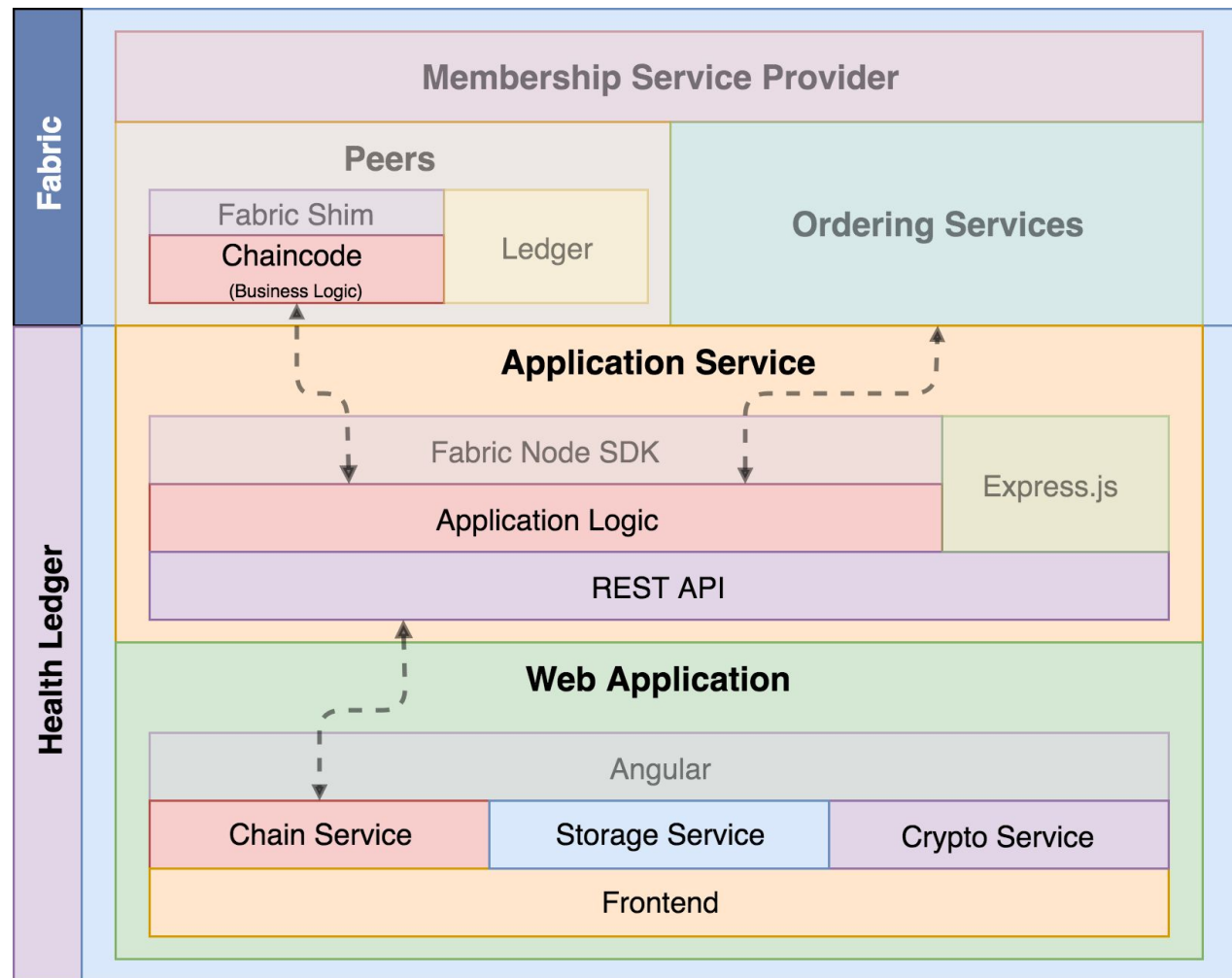
Datenmodell



Architektur



Architektur



Technologien

Frontend

- Programmiersprachen
 - Typescript
- Frameworks / Libraries
 - Angular 6
 - Angular Material Design

Backend

- Programmiersprachen
 - Javascript
- Frameworks / Libraries
 - NodeJS
 - Fabric Node SDK
 - Express.JS

Distributed Ledger

- Programmiersprachen
 - JavaScript
- Frameworks / Libraries
 - Hyperledger Fabric
 - NodeJS
 - Fabric Shim

DEMO

Codebeispiele

Lessons Learned

Hyperledger:

- ursprünglich getrennt entwickelte, sich überlappende Teilprojekte
- die meisten Projekte nutzen GitHub nur read-only
 - erschwert Kollaboration

Hyperledger Fabric:

- Fabric-Client-SDK unsaubere und fehlerhafte Implementierung
 - Fehlerbehandlung bei langlebigen TCP Verbindungen
 - Konfiguration im node_modules Ordner
 - Speichert zur Laufzeit Zertifikate im node_modules Ordner
- Unzureichende bis keine Entwicklungsumgebung für Chaincodes

Lessons Learned

Frontend:

- Effektive Angular-Entwicklung, allerdings lange Einarbeitung
- “Code Splitting”-Konzept durch Component Router ermöglicht Aufgabenteilung
- Angular Material leicht einzubinden

Lessons Learned

Hyperledger Explorer:

- Entwicklung nicht synchron mit Fabric, lückenhafte Dokumentation

In-Browser Verschlüsselung:

- keine offensichtlichen Lösungen
- Projekte wenig maintained
- Diskussion ECC vs. RSA

Dokumentation:

- Laufender Prozess
- Interne Absprachen müssen festgehalten werden
- es hilft, wenn die Dokumentation immer aktuell ist

Vielen Dank!
Fragen?



Health Ledger

Die Digitale Patientenakte.

Cem Basoglu, Mario Cichonczyk, Matthias Kersting, Nils Kirchhof, Fynn Klöpper, Kevin Schima, Patrick Starzynski