



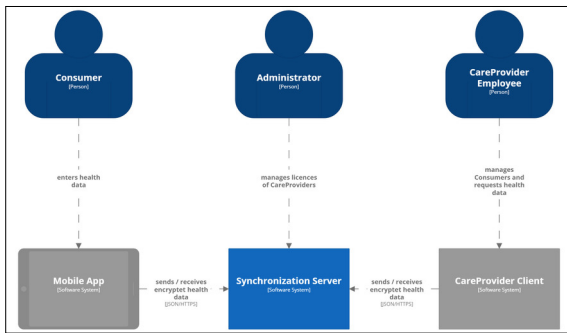
Moritz Habegger



Micha Schena

|              |   |
|--------------|---|
| Diplomanden  | Moritz Habegger, Micha Schena                           |
| Examinator   | Prof. Dr. Olaf Zimmermann                               |
| Experte      | Dr. Gerald Reif, Innovation Process Technology, Zug, ZG |
| Themengebiet | Internet-Technologien und -Anwendungen                  |

## Cloud-Native Refactoring in einem mHealth Szenario

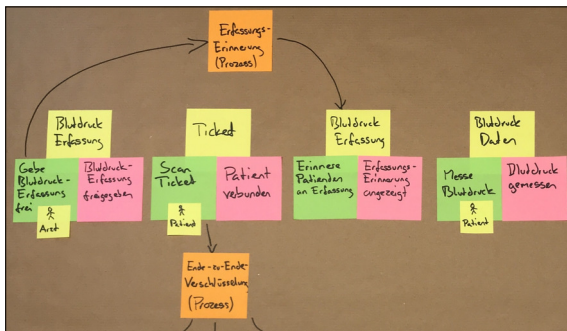


C4 System Context Diagram – Kommunikationslösung  
Eigene Darstellung

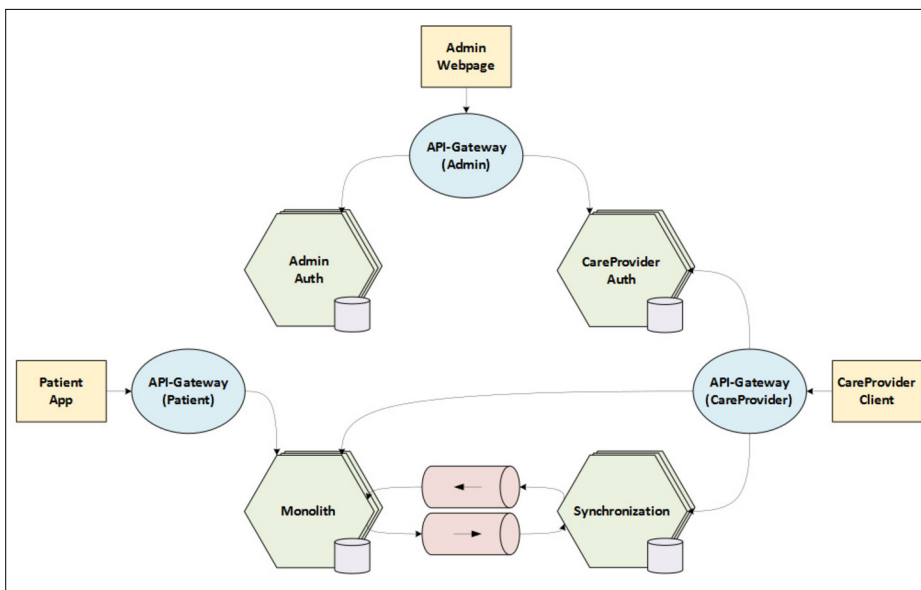
**Ausgangslage:** Die Bachelorarbeit untersucht eine bestehende Spring Boot Integrationslösung, die einen Ende-zu-Ende verschlüsselten Kommunikationskanal zwischen Gesundheitsdienstleistern und deren Patientengut anbietet. Diese Applikation weist eine monolithische Architektur auf, die mithilfe eines cloud-native Refactorings auf eine Microservice-Architektur flexibilisiert werden soll. Microservices sind ein branchenübergreifender Software-Architektur-Trend. Die Arbeit soll zeigen, ob und wie ein solcher Umbau in diesem Szenario möglich ist und was die Konsequenzen hinsichtlich Wirtschaftlichkeit, Stabilität und Flexibilität sind.

**Vorgehen:** Im Rahmen der Analyse wurden die funktionalen sowie nicht funktionalen Anforderungen an die umzubauende Applikation erfasst und die bestehende Implementierung mit der Modellierungstechnik C4 dokumentiert. Ein Event Storming lieferte eine Geschäftsprozess-Modellierung, die die Anforderungen tiefer erfasst und auch auf konkrete Bounded Contexts aus dem Domain-Driven Design hinweist, die dann wiederum in Microservices abgebildet werden können. Das erarbeitete Verständnis für die Applikation führte zu mehreren Microservice-Architekturvorschlägen, deren Eignung systematisch evaluiert wurde. Der letzte Abschnitt der Arbeit setzte den ausgewählten Architekturvorschlag prototypisch mit Spring Cloud und RabbitMQ um.

**Ergebnis:** Die monolithische Applikation wurde erfolgreich zu autonomen Microservices umgebaut. Das entwickelte und erprobte Vorgehen hat sich als effizient und zielführend erwiesen und ist so gewählt, dass es sich auch auf andere Branchen und Szenarien übertragen lässt. Gewisse Architektur-Entscheidungen in der bestehenden Applikation haben den Umbau zu Microservices erschwert, welche in einem neuen Projekt vermieden werden sollten. Eine Ergebnisdiskussion stellt die Vor- und Nachteile, beziehungsweise die Konsequenzen der umgesetzten Architektur kritisch gegenüber.



Ausschnitt aus dem Event Storming Workshop  
Eigene Darstellung



Umgesetzte Architektur des Prototyps  
Eigene Darstellung