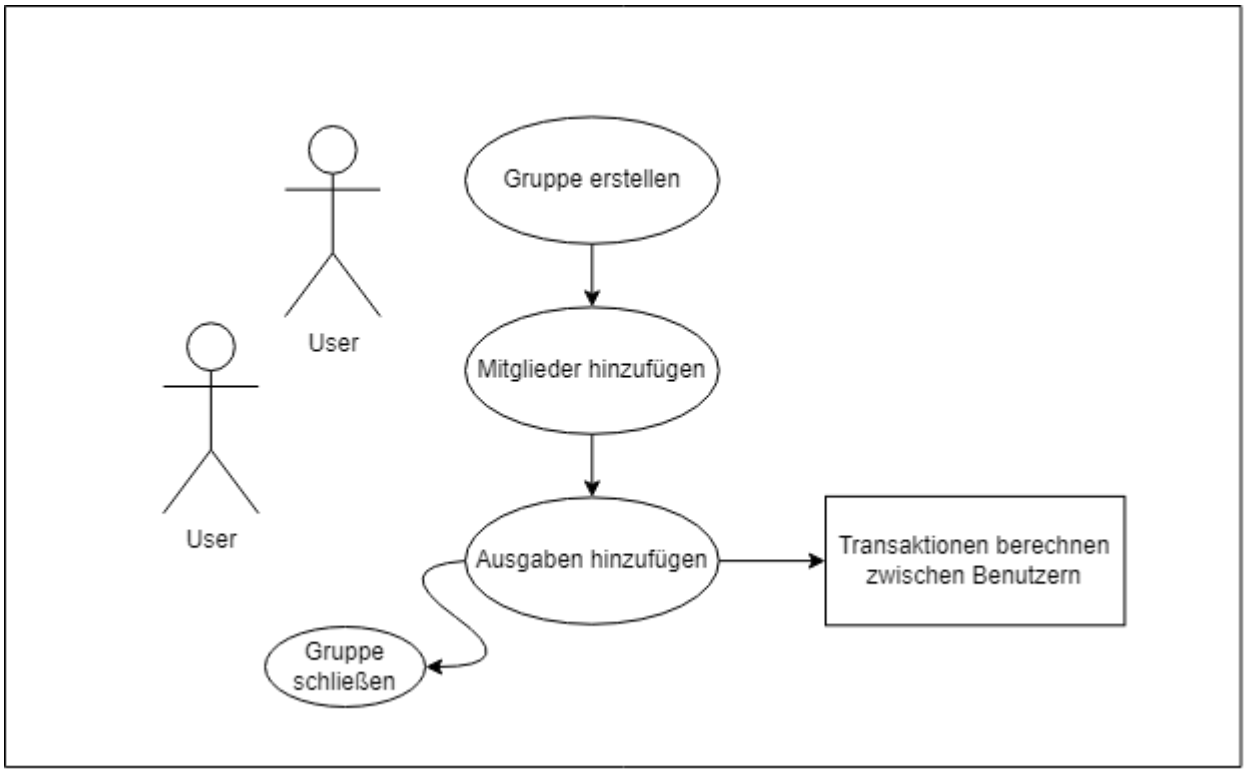
Arc42-Dokumentation

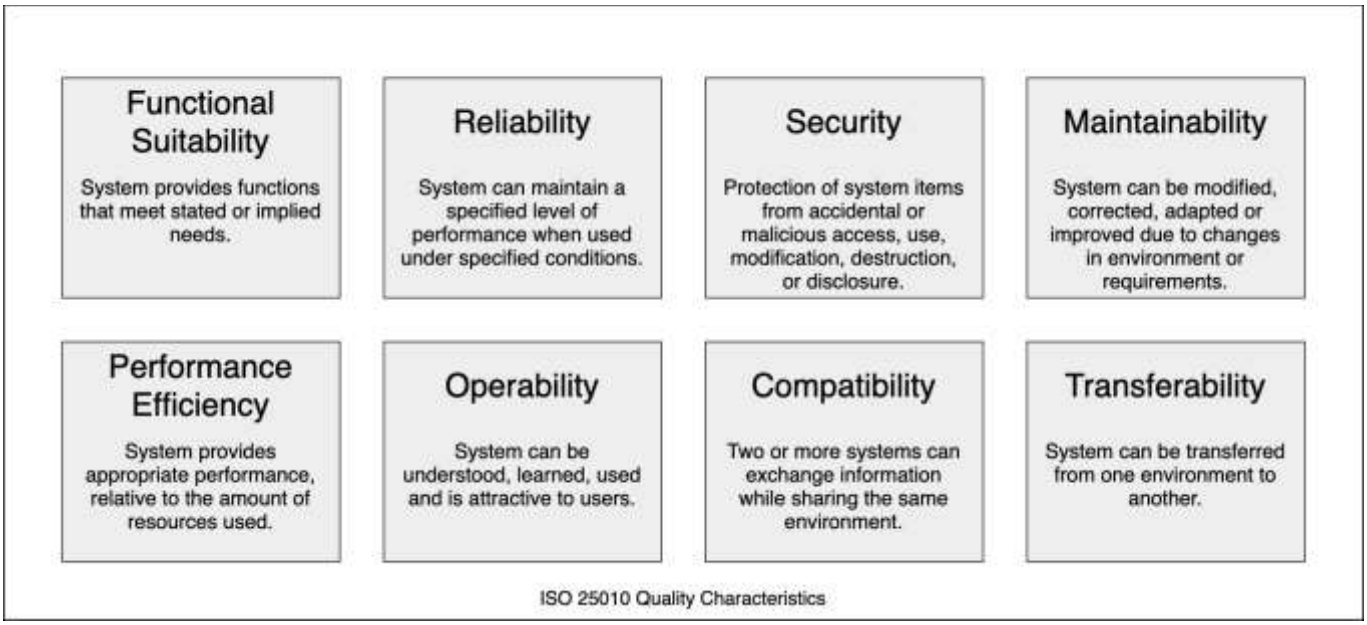
Einführung und Ziele

Aufgabenstellung

Minimal-Viable-Product, welches Transaktionen in einer Gruppe berechnet. Die melden sich mit deren Github-Konten an. Jede Person kann eine Gruppe erstellen, innerhalb dieser Gruppen kann man weitere Person hinzufügen und Ausgaben erstellen, welche in der Transaktionenseite minimal berechnet werden.



Qualitätsziele



Laufzeitsicht

Laufzeit-technisch ist das größte Problem die minimale Berechnung der Transaktionen, da dieses Problem NP-schwer ist. Die Herangehensweise ist bei der Implementierung unseres Services exponential, welches zu einem Minimal-Subset-Problem umgeformt wurde, wobei alle Subsets berechnet werden die möglich sind.

Randbedingungen

Technische Randbedingungen

Programmiersprache: Java
Framework: Spring Boot
Build-System: Gradle
Programmanalyse: SpotBugs, Checkstyle
Konventionen: Google Style Guide, Java Konventionen

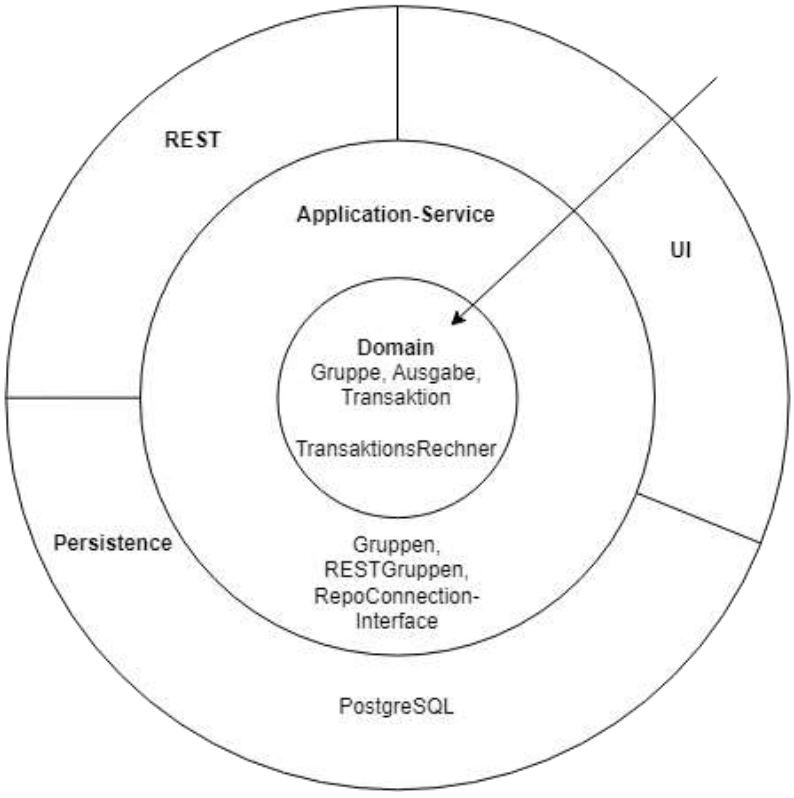
Organisatorische Randbedingungen

Wir haben uns in zwei Teams aufgeteilt: Frontend, Backend. Vor der Aufteilung haben wir Aspekte wie Checkstyles, SpotBugs, Docker und ArchUnit-Tests erledigt. Außerdem haben wir das Domänen-Modell und die Schnittstelle (Service-Interfaces) konstruiert, damit die Teams parallel arbeiten können.

Service-Interfaces: Transaktionsrechner, Gruppen und später RESTGruppen

Querschnittliche Konzepte

Onion-Architektur



Domain Modell

Gruppe → ID, Titel, eine Menge von Teilnehmern, eine Menge von Ausgaben und Schließungsstatus der Gruppe

Ausgabe → Titel, Betrag, Gläubiger, eine Menge von Teilnehmern und Datum der Ausgabenerstellung

Transaktion → Schuldner, Gläubiger, Betrag

Die Transaktion stellt den Ausgleich der Verbindlichkeit dar.

Lösungsstrategie

TransactionCalculator

Benötigt:

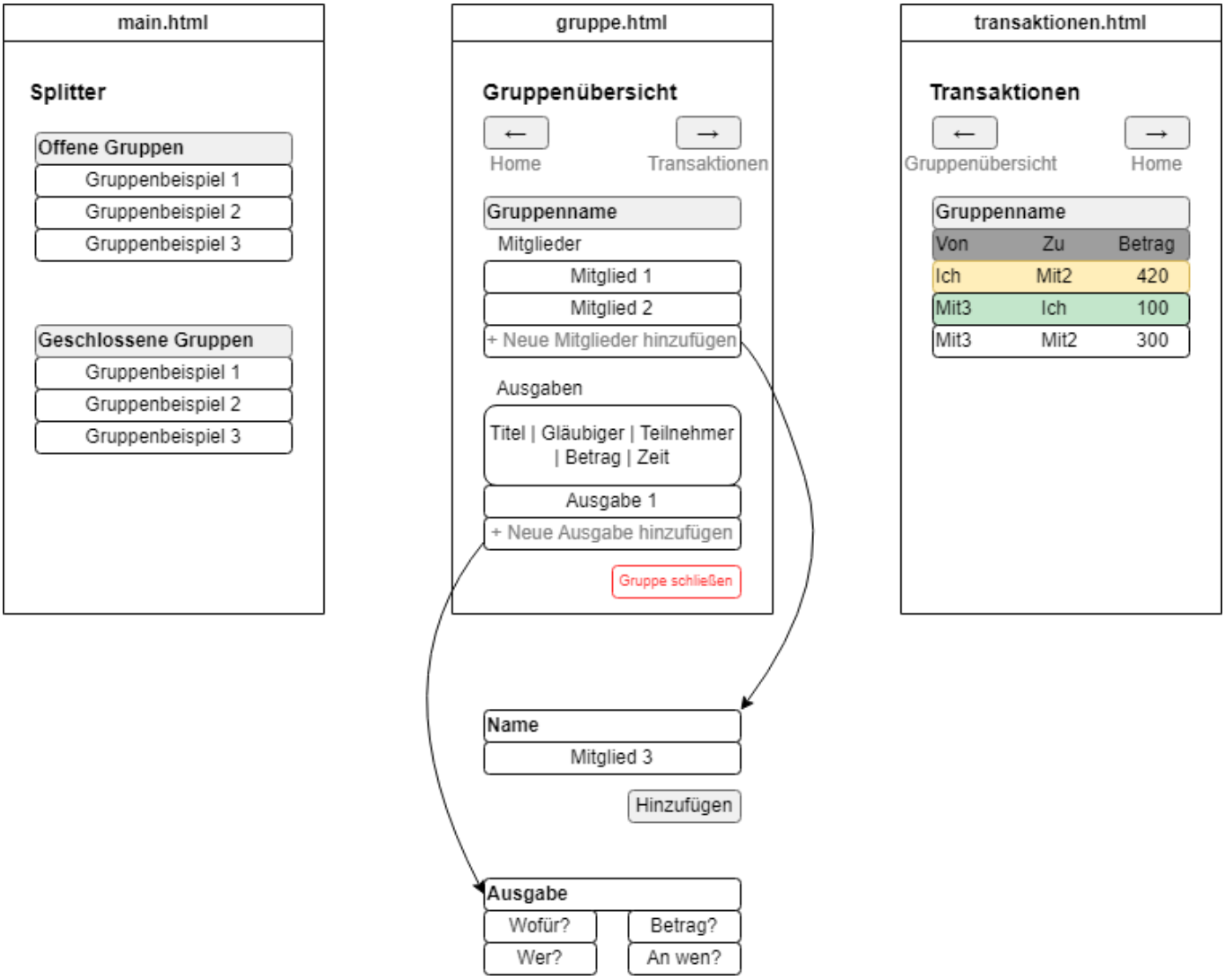
SubsetCalculator und Liabilities

Algortihmus

Im Folgenden wird einfach nur von Darlehen gesprochen um den Algorithmus verständlich zu erklären. Der Algorithmus funktioniert so jedoch nur, wenn die Anzahl an Gläubigern weniger ist als der Anzahl an Schuldnern. Wenn es also mehr Schuldner gibt, tauschen wir die Mengen und versuchen die Schulden auszugleichen (die kleinere Menge wird also immer ausgeglichen).

1. Berechne Verbindlichkeiten → Ist man insgesamt im Plus oder Minus?
2. Teile in Liste mit positiven und negativen Verbindlichkeiten
3. Wähle kleinstes Darlehen
4. Berechne alle Teilsummen der Schulden die dem Darlehen entsprechen
    a. Falls mehrere Teilsumme existieren wähle die mit der kleinsten Anzahl an Transaktionen
        i. Zurück zu Schritt 3
    b. Falls keine existiert Schritt 5
5. Verwende die Größten Schulden um Auszugleichen
    a. Ggf. mehrere Schulden

Architekturentscheidung



Qualitätsanforderungen

Das System sollte immer möglich seine Ausgaben in minimale Transaktionen umzuformen unter der Bedingung der gegebenen Laufzeit. Das System ist geschützt durch die Springboot-Security und OAuth2-Security, wodurch keine interne Logik sichtbar ist. Spotbugs ist ein weiteres Feature für die Beibehaltung des Information-Hiding-Principle. Die Wartbarkeit wird gewährleistet, da SOLID eingehalten wird und die Architektur für einfache Erweiterungen sorgt.

Glossar

Begriff	Definition
Gruppenmitglied	Gruppenmitglied kann sowohl eine Gruppe erstellen oder zu einer Gruppe hinzugefügt werden
Ausgabe	Ausgabe beläuft sich zwischen Gläubiger und Schuldnern
Verbindlichkeit/ Liability	Berechnung, ob man verschuldet ist oder man schuldet
Transaktion	Ausgleich der Verbindlichkeit