

## Selbstreflektion von Dimitri Dorn

Bei der Erarbeitung dieser Hausarbeit konnte ich die grundlegenden Funktionalitäten, die das Spring Framework bietet, kennenlernen. Da in diesem Projekt mein Aufgabenbereich in der Backend Entwicklung lag, musste ich mich besonders viel mit der Erstellung und Nutzung der Datenbanken, sowie mir der Logik der API-Endpoints auseinandersetzen. Zudem konnte ich Erfahrungen in der SCRUM Entwicklung sammeln.

Der Vorteil des Spring Frameworks bestand hauptsächlich in dem Zusammenspiel zwischen den API-Endpoints und dem JPA-Service mit dem Hibernate ORM-Framework. Durch das Instantzieren des JPA Repositorys fiel es sehr einfach, die übergebenen Daten aus den API-Endpoints in die Datenbank zu integrieren, da die entsprechenden Datenbankintention mit einem intuitiven Funktionsnamen durchgeführt werden kann. Auch bei der Selektion der Daten erwies sich das JPA Repository als ein profitables Tool, da durch die vordefinierten Selektionsmethoden keine SQL-Querys mehr definiert werden mussten.

Allerdings erwiesen sich auch Nachteile in diesem Framework. Der erste Punkt besteht darin, dass sich die Erstellung der Tabellenverknüpfungen als äußerst mühselig erwies. Die Schwierigkeit lag hierbei bei der Syntax, die von Spring verwendet wird, um eine Verknüpfung auf die einzelnen Entität-IDs zu erstellen. Außerdem beinhalten die Relationsannotationen eine große Anzahl an Einstellparameter, die die Übersicht über die Relationsmechanik verkomplizieren. Eine weitere Problematik tauchte bei der Einarbeitung in das Spring Framework auf, wenn die Webapplikation in der Sprache „Kotlin“ geschrieben werden soll, da sich die meisten Syntax- und Annotations-Definitionen auf den Java-Fall beziehen.

Design Patterns wurden kaum angewandt. Grund dafür ist, dass durch die Spring Annotationen klare Struktur für die Zuständigkeiten entstand. So wurde zum Beispiel für Klassen mit der `@RestApi` Annotation die ganze Endpointlogik ausgelagert. Um in der Datenbank auf bestimmte Daten zuzugreifen oder diese zu verändern, wurde für jede Tabelle ein Repository mit der `@Repository` Annotation erstellt. Die Validierung und Prüfung der empfangenen Daten wurden in `@Service` Klassen durchgeführt.

Wie am Anfang erwähnt, wurde in der Gruppe SCRUM als Vorgehensmodell gewählt. Das erwies sich als sehr vorteilhaft, weil auf Änderungen schnell eingegangen werden konnte und auch entstandene Probleme/Bux rechtzeitig beseitigt werden konnten.