**Deploymentstrategie**

Um in hoher Frequenz unseren Kunden neue Features liefern zu können Entwicklern wir eine einfache Deploymentstrategie, um unseren Code einfach, schnell und möglichst automatisiert auf Produktions- oder Testumgebungen zu bekommen. Dadurch soll zum einen das Entwickeln viel Zeit und Arbeit gespart werden, welche für andere Aufgaben genutzt werden kann. Des Weiteren wird dadurch auch die Produktqualität verbessert, da neuer Code direkt unter „realistischen“ Bedingungen getestet werden kann.

**Vorgehen:**

* Containerisierung mit Docker: Alle eigenständigen Komponenten des Systems (Frontend, Backend und Datenbank) sollen in separate Container gepackt werden. Dies ermöglicht eine konsistente Umgebung angepasst für die jeweilige Komponente. Mithilfe eines Docker Compose können die Container dann verwalten werden
* CI/CD-Pipeline: Mithilfe von Github Actions soll eine einfache CI/CD-Pipeline erstellt werden. Die Pipeline zieht dann automatisch den Code von GitHub baut die einzelnen Komponenten und packt diese in Container. Hier können auch noch Test eingebaut werden, welche den Code überprüft.
* Cloud-Plattform: Am Ende der Pipeline können die Container dann auf einer Cloud-Infrastruktur bereitgestellt werden, die speziell darauf ausgelegt ist, Container zu hosten und zu verwalten.

**Regeln:**

* Der Code auf dem main Branch landet auf der Testumgebung
* Der Stable-State Branch wird alle zwei Wochen nach dem Review auf eine Produktionsumgebung deployt. Wenn der Code von main beim Review abgenommen wurde, wird dieser in den Stable-State Branch gemerged.