

Projektarbeit Aquaponik

**Anforderungsanalyse und Umsetzung einer Software-Lösung im
Bereich Aquaponik am Beispiel der ZHAW Wädenswil**

Deniz Akca

Dennis Bannerman

Mike Iten

ZHAW - Zurich

18. Dezember 2021

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ausgangslage	1
1.2	Aufgabenstellung	1
1.3	Zielsetzung	1
2	Grundlagen	1
2.1	Hosting	1
2.2	Backend Technologie	1
2.3	Schnittstelle	1
2.4	Frontend Technologie	2
2.5	Datenbank	2
3	Vorgehen / Methoden	2
3.1	Swagger	2
3.2	Continuous Integration	2
3.3	Docker	3
3.3.1	Docker-Image	3
3.3.2	Docker-Compose	3
4	Resultate	3
5	Diskussion und Ausblick	3
5.1	Sprint 0 Meeting	3
5.2	Sprint 1 Meeting	3
5.3	Sprint 2 Meeting	4
6	Verzeichnisse	4
6.1	Literaturverzeichnis	4
6.2	Glossar	4
6.3	Abbildungsverzeichnis	4
7	Anhang	4

1 Einleitung

1.1 Ausgangslage

Ein Aquaponics Projekt besteht bereits aus folgenden Komponenten. Sensoren der Aquaponics Systeme, welche an SC1000 Geräten angeschlossen sind. Ein serieller Bus verknüpft alle SC1000 mit einem RasPi welches über eine Modbus-API die Sensordaten auf eine MySQL Datenbank ablegt. Diese Daten werden auf der Webseite dargestellt.

Die Webseite, welche unter myaquaculturefarm.ch zu finden ist, wird auf hosttech.ch gehostet. Hosttech verwendet als Backend Technologie PHP. Unsere Konfigurationsseite wird ebenfalls auf dieser Domain parallel zu den anderen Webseiten von ZHAW Life Sciences und Facility Management gehostet.

Beim Backend sind wir gebunden was die Host-Firma uns zur Verfügung stellt, in diesem Falle wäre das PHP.

1.2 Aufgabenstellung

Die Datenbank, in der die Sensordaten geloggt werden, besteht aus zwei Tabellen. In einer der Tabellen werden die Sensoren eingetragen, die sich in den Systemen befinden und in der zweiten Tabelle werden die Sensordaten abgespeichert und mit dem jeweiligen Sensor verknüpft.

Das Bearbeiten dieser Zuordnungstabelle ist für die Mitarbeiter/Studierende der ZHAW Life Sciences und Facility Management mit dem von Hosttech gegebenen Tool «phpMyAdmin» nicht verständlich. Zusätzlich müssen spezifische Werte eingegeben werden die einen Informatik Laien nicht bekannt sind, welches zu inkorrekte Angabe von Daten führen kann, welches wiederum zu einem Durcheinander in der Log-Tabelle führt. Das «phpMyAdmin» Tool ist ebenfalls nur per Verwaltungsseite der Hosttech Domain erreichbar welches eine zusätzliche Hürde darstellt.

1.3 Zielsetzung

Um die Zuordnungstabelle einfacher zu bearbeiten, soll der Ablauf abgändert werden. Als Lösung stellen wir eine REST Schnittstelle zur Verfügung über welche die Tabelle mit wenigen Handgriffen verändert werden kann. Diese soll übersichtlich und einfach zu bedienen sein. Damit die Schnittstelle zu jeder Zeit erreichbar ist soll sie gehostet werden.

2 Grundlagen

2.1 Hosting

Unser Produkt soll auf dem neusten Stand der Dinge sein und daher verzichten wir auf den Hoster Hosttech der schon für das bestehende System verwendet wird. Ein grosser Nachteil von Hosttech ist das wir auf ein PHP Backend beschränkt sind. Um auch das Deployment zu modernisieren und hauptsächlich zu automatisieren möchten wir mit Dockercontainer arbeiten. Hierbei hat unser Betreuer seine Plattform angeboten. Auf dieser können wir ein Image des Dockercontainers hosten lassen. Somit ist dieses zu jederzeit erreichbar. Dies gilt für das Frontend sowie für das Backend. Damit der Server nach der Umsetzung nach wie vor erreichbar ist, wurde uns vom Betreuer folgende Seite vorgeschlagen: <https://www.hetzner.com/de/>

2.2 Backend Technologie

Mit dem Loslösen von Hosttech sind wir frei die Backend Technologie zu bestimmen und wir haben uns auf Java/Spring-Boot geeinigt. Wir haben uns für Spring-Boot entschieden, da alle Entwickler mit Java vertraut sind und OpenAPI Spring-Boot unterstützt. Spring-Boot eignet sich auch gut für unsere Aufgabenstellung, da wir eine Datenbank anschliessen werden und die Daten auf der Webseite nicht nur dargestellt werden sondern auch manipuliert werden können, eignet sich Spring-Boot mit der einfachen Handhabung von JPA gut.

2.3 Schnittstelle

Als Schnittstelle zwischen Backend und Frontend wollen wir das State of the Art Tool OpenAPI einsetzen. Nach genauerer Analyse sehen wir einen grossen Nutzen des Tools, da es uns eine grosse Unterstützung zur Entwicklung der REST Schnittstelle sein wird. In OpenAPI

kann in einem YAML File die ganze REST Schnittstelle erstellt und konfiguriert werden daraus kann dann OpenAPI unser Spring-Boot Backend erzeugen was uns viel Arbeit abnimmt.

2.4 Frontend Technologie

Als Frontend-Framework sind wir frei, da es keinen Wunsch/Voraussetzung vom Kunden gab. Wir als Gruppe haben uns für Angular(2+) entschieden, da die Mehrheit der Gruppe sich mit diesem Framework gut auskennt.

2.5 Datenbank

Die bestehende Datenbank auf Hosttech wird in einen neuen Docker-Container migriert und neben dem Spring-Boot Projekt gehostet. Nur das Spring-Boot Projekt hat direkten Zugriff auf die Datenbank, alle Änderungen müssen über die REST Schnittstelle der Spring-Boot Applikation erfolgen.

Nach Analyse der Datenbank sahen wir ebenfalls Potential für Verbesserungen, doch wollten wir nicht zu viel an der Struktur ändern, da diese Änderungen weitere Anpassungen auf Systemen erfordern, welche wir keinen Einfluss nehmen können, daher haben wir uns entschieden die Datenbank zu belassen und eine 1 zu 1 Kopie zu erstellen.

3 Vorgehen / Methoden

3.1 Swagger

Erste Umsetzungen finden mithilfe eines Swaggerprojekts statt. Hier kann ein Entwurf erstellt werden. Mithilfe dieses Entwurfs wird ein Grundgerüst hergestellt, welcher dazu dient, vereinfacht die REST Schnittstelle zu bauen.

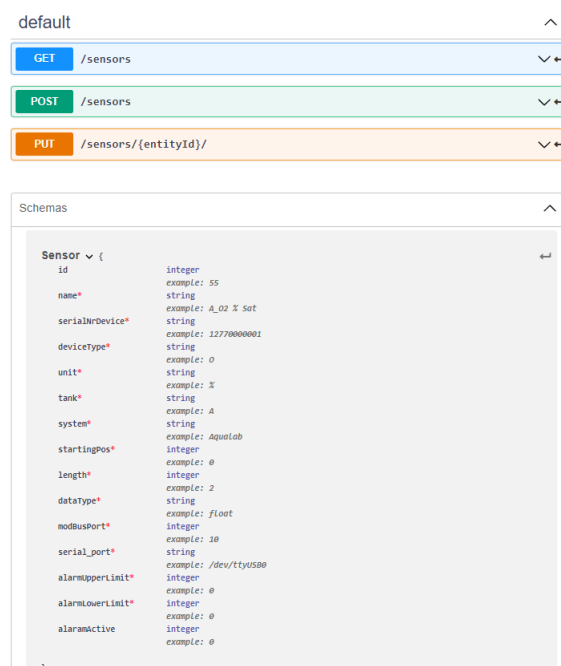


Abbildung 1: Swagger Beispiel

Die Schnittstelle soll dazu fähig sein Daten der Sensoren zurückzugeben. Die Möglichkeit soll bestehen, dass die Sensorendaten zusätzlich editierbar sind.

Als Vereinfachung können die Sensoren auch als eine Liste angezeigt werden.

3.2 Continuous Integration

Als Source Control wird Git eingesetzt und um mit allen Entwicklern zusammen zu arbeiten befindet sich das Repository auch auf github.com. Github bietet uns mit den Github Actions ein gutes Tool welches wir automatisch das gesamte Spring-Boot Projekt erstellen, testen und auf DockerHub hochladen können. Die Github Actions werden bei jedem Push in den Main Branch angestoßen. In den Main Branch wird in der Regel nur dann gepusht wenn ein

Feature Branch nach Pull Request und Code Review in den Main gemergerd wird. Daher wird das Github Actions nicht bei jedem Commit ausgeführt und nur funktionierende Versionen werden auf Dockerhub hochgeladen.

3.3 Docker

Docker kann sehr vielfältig eingesetzt werden, aber worin Docker seine Stärken hat ist im Deployment. Mit einem gutem Docker-Compose File ist ein komplexes Multi-Container System mit einem Befehl aufgesetzt.

3.3.1 Docker-Image

Von unserer Spring-Boot Applikation erstellen wir ein Image welches auf DockerHub hochgeladen wird. Von der Datenbank erstellen wir kein Image da DockerHub wie Github öffentlich zugänglich ist und wir die Datenbank nicht veröffentlichen wollen. Die Datenbank wird dann beim erstellen des MySQL Containers eingelesen wie im nächsten Abschnitt erklärt.

3.3.2 Docker-Compose

Das Docker-Compose holt sich die neueste Version der Applikation und erstellt ein Netzwerk für das gesamte PAAQ-System. Der Applikation wird ein Portforwarding eingerichtet, dass diese auch von aussen zugänglich ist. Anschliessend wird auch ein MySQL-Container heruntergeladen und in das gleiche Netzwerk eingehängt, dass eine Kommunikation zwischen Spring-Boot Applikation und MySQL-Datenbank stattfinden kann. Der MySQL Datenbank wird noch der Dump der alten Datenbank mitgegeben, dass eine exakte Kopie der vorherigen Datenbank erstellt.

4 Resultate

5 Diskussion und Ausblick

5.1 Sprint 0 Meeting

- Angular Demo von Herr Bachmann TodoAngular.
 - In Github Frontend - Package.JSON ist alles vorhanden was zu diesem Thema als Wissen benötigt wird.
- Hosting kann von Herrn Bachmann übernommen werden.
 - Hierfür muss entweder ein Dockercontainer für das Frontend und Backend respektive erstellt werden, oder für das Frontend und Backend kombiniert.
- Cloud Service für VM Möglichkeiten.
 - Auf der Homepage <https://www.hetzner.com> gibt es zahlreiche kosteneffiziente und interessante Angebote für verschiedene Services.
- Erstellen und diskutieren der Tasks für Sprint 1.

5.2 Sprint 1 Meeting

- Durch verschiedene Versionierungen und unterschiedlichen Arbeitsstationen kam es zu einigen Problemen mit Gradle unter anderem auch Dependenciesfehler.
- Unter längerer Diskussion wurde beschlossen dass das Frontend und Backend in einem Dockercontainer kombiniert werden um das hosten und arbeiten zu vereinfachen.
- Eine kurze Dockerpräsentation auf einer virtuellen Ubuntumaschine über den aktuellen Stand wird vorgeführt.
- Angular Frontend steht noch offen und wird als Letztes angegangen.
- Ein weiterer Punkt wäre ein mögliches Treffen mit dem Team aus Wädischwil bezüglich Mockups. Um eine grobe Richtung für ungefähre GUI Vorstellungen zu bekommen oder ob dies komplett nach eigenem Ermessen erstellt werden soll.

5.3 Sprint 2 Meeting

- Um weiter frotzufahren mit dem GUI
- Erklärung Docker Benutzung
- Andere Hostverwendung (Hetzner)
- Tabellarisch für Einstellungen der Sensoren
- Reihenfolge gleich wie bei SC1000 geregelt
- Adressen der Sensoren werden im SC1000 geregelt
- Daten werden über MQTT zur Urbanblue geschickt (PHP)
- Zeitfenster 2 Wochen bis Frontend fertig

6 Verzeichnisse

6.1 Literaturverzeichnis

6.2 Glossar

6.3 Abbildungsverzeichnis

7 Anhang