Richtlinie zum „Software-Sprint“

SearchWing – Darstellung von Drohnendaten im Dienst der Seenotrettung

Schlussbericht

Zuwendungsempfänger:

Mai Ly Le & Ricardo Martinez Garcia GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS23S24 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Jedes Jahr wagen sich Tausende von Menschen auf eine gefährliche Überfahrt über den Ozean, in der Hoffnung, auf dem europäischen Kontinent ein besseres Leben zu finden. Das Fehlen einer koordinierten Herangehensweise und politische Uneinigkeiten zwischen den EU-Ländern führen dazu, dass Tausende von Menschen jedes Jahr unnötig ihr Leben im Mittelmeer verlieren. Als Reaktion auf diese Tragödie setzen verschiedene private Organisationen ihre eigenen Ressourcen ein, um die Situation zu entschärfen. Das Projekt SearchWing wurde im Jahr 2016 ins Leben gerufen, um diese Organisationen mit autonomen Drohnen für die Ortung von Menschen in Not auszustatten.

Die Entwicklung und der Betrieb von Drohnen sind komplexe Aufgaben, die mit der Erfassung und Verarbeitung einer Vielzahl von Daten während der Einsätze einhergehen, darunter Sensordaten der Drohne, Bilder und Wetterbedingungen. Die Analyse dieser Daten ist von entscheidender Bedeutung, um den Betrieb der Drohnen effizienter und sicherer zu gestalten. Um die Herausforderungen im Bereich der Datenverwaltung und Datenqualität zu bewältigen, haben wir den LogViewer entwickelt. Dabei handelt es sich um eine webbasierte Anwendung, die dazu dient, Flugdaten nach den Einsätzen zu kategorisieren, sie zu verarbeiten, um Flugstatistiken und -parameter zu extrahieren, sowie Sensorzeitreihendaten grafisch darzustellen.

Beitrag des Projektes zu den Zielen der Förderinitiative „Software-Sprint“

Die Hauptzielgruppe SearchWings sind Mitglieder von NGOs, die humanitäre Einsätze im Mittelmeer und im Atlantik unterstützen. Im Gegensatz zu staatlichen/militärischen Organisationen, die im gleichen Bereich tätig sind, werden NGOs ausschließlich von Privatpersonen geführt. Die Ziele SearchWings und dementsprechend auch der Zweck der LogViewer-App lassen sich im Bereich "Civic Tech" verordnen. Denn bei der LogViewer App handelt es sich um eine Technologie, die von Aktivististis entwickelt wurde, um die Arbeit anderer Aktivististis im Feld der Seenotrettung zu unterstützen, befähigen und zu erleichtern.

Das Hauptziel der App besteht darin, den Nutzern Einblicke in Drohnenflüge zu ermöglichen. Mit Hilfe der LogViewer-Anwendung können Drohnenbetreiber nachverfolgen, was bei ihren Einsätzen schief gelaufen ist, wohin die Drohne genau geflogen ist oder wie viel Akku verbraucht wurde. Andernfalls müssten Drohnenbetreiber auf komplexe Tools oder Skripte zurückgreifen, die keine benutzerfreundliche grafische Benutzeroberfläche bieten und daher nicht für jeden Nutzer zugänglich sind. Infolgedessen trägt der LogViewer auch zur Förderung der "Datenkompetenz" bei, indem die Anwendung Drohnendaten verständlicher und nachvollziehbarer macht.

Ausführliche Darstellung der Ergebnisse

Während der Förderphase lag unser Schwerpunkt auf der Entwicklung eines Proof of Concept (PoC) für die Web-Anwendung LogViewer, die zur Darstellung und Verwaltung von Drohnendaten dient.

In der ersten Phase führten wir eine Explorationsphase durch, in der wir gemeinsam mit Mitgliedern des SearchWing-Projekts die Anforderungen an die Anwendung definierten und das UI-Design entwickelten. In dieser Phase legten wir auch die Architektur der Anwendung fest und untersuchten verschiedene Kommunikationsmöglichkeiten zwischen Backend und Frontend, darunter REST und GraphQL. Am Ende des ersten Finanzierungsmonats hatten wir ein klares Verständnis dafür, wie die Anwendung funktionieren würde, wie die Dienste miteinander interagieren würden, wie Benutzer mit ihr umgehen würden und welches Datenformat und welche Datenmenge verarbeitet werden müssten. Zudem erstellten wir einen Arbeitsplan für den Rest der Finanzierungsphase.

In der zweiten Phase konzentrierten wir uns auf die Entwicklung grundlegender Funktionen der App. Dazu gehörten verschiedene Tabellenansichten, in denen Benutzer die gesamte Drohnenflotte, Missionen und Flüge verwalten und Dateien für jeden Flug hochladen konnten. Die wichtigsten Meilensteine in dieser Phase waren:

1. Entwurf und Entwicklung wiederverwendbarer UI-Komponenten für Tabellenansichten, die Flüge, Missionen und Drohnen darstellen und Sortierungs- sowie Filterungsfunktionen bieten.

2. Definition des Datenbankschemas und der API-Endpunkte für CRUD-Operationen.

3. Festlegung gemeinsamer Datentypen und der Verbindung zwischen Backend und Frontend.

4. Einrichtung der Infrastruktur, einschließlich Test- und Release-Pipelines sowie Bereitstellung von Servern für Staging-Umgebungen.

Nach der Implementierung der ersten Features erstellten wir unsere Produktionsumgebung, in der Benutzer bereits echte Daten hochladen konnten. Dies brachte zusätzliche Herausforderungen mit sich, insbesondere aufgrund der enormen Größe der Flugdaten. Speicherplatz kann teuer sein, daher übernahmen wir die Installation und Konfiguration des Datenbankservers selbst. Neben den vorhandenen Servern mussten wir zusätzlichen Speicherplatz erwerben und mit unserer Datenbank verknüpfen. Leider gab es nur begrenzte Dokumentation zur Integration von neuem Datenspeicher, was zu Verzögerungen führte.

Der letzte Teil des Projekts konzentrierte sich auf die Entwicklung von Funktionen zur Verarbeitung von Dateien und zur Darstellung von Zeitreihendaten. Das Endergebnis war ein Plot Viewer, der es Benutzern ermöglichte, die Daten eines einzelnen Fluges darzustellen und einen anderen Flug auszuwählen, um Vergleiche anzustellen. Hierin liegt der Hauptnutzen der App, da es keine bestehende Anwendung gibt, die solche Flugvergleiche ermöglicht. Dies brachte einige Herausforderungen mit sich, wie die Entwicklung eines geeigneten Verfahrens zur Panning-Funktion im Frontend.

Zusätzlich mussten wir unsere eigene Python-Bibliothek entwickeln, um die Daten auf dem Server effizient zu verarbeiten. Aufgrund der großen Datenmenge gab es auch Leistungsprobleme im Frontend, daher mussten wir unsere Architektur und die Art und Weise, wie die Kommunikation zwischen Backend und Frontend erfolgte, neu gestalten. Dieser Teil war zweifellos der anspruchsvollste, brachte jedoch die wertvollsten Erkenntnisse:

1. Die Verwaltung von Zeitreihendaten erfordert eine serverseitige Vorverarbeitung.

2. Das Frontend sollte dynamisch entscheiden, wie viele Daten geladen werden, abhängig von der Pixelbreite des angezeigten Bereichs der Zeitreihen.

3. Bei der Arbeit mit Drohnen-Logdateien ist eine sorgfältige Speicherverwaltung unerlässlich, da die Dateien sehr groß sind.

Am Ende der Finanzierungsphase hatten wir einen funktionierenden Prototyp der Anwendung entwickelt.

Der Prototyp umfasst folgende Seiten und Funktionen:

1. Flight-Page, Misssion-Page und Drone-Page:

Diese Seiten bieten tabellarische Ansichten für Drohnen, Missionen und Flüge, um die in der Datenbank gespeicherten Objekte darzustellen. Die Tabellen dienen gleichzeitig als Formulare, über

die Benutzer Eigenschaften der gespeicherten Flug-, Missions- und Drohnenobjekte ändern oder löschen können. Außerdem ermöglichen sie die Suche und Sortierung der Objekte.

2. Add-Page für Mission, Flight und Drone:

Diese Seiten enthalten jeweils Formulare, um neue Missionen, Drohnen und Flüge hinzuzufügen. Im Formular für Flüge können Benutzer auch die entsprechenden Log-Dateien hochladen.

3. Plot Viewer bzw. Detail-Page eines Fluges:

Wenn ein Flug über eine Log-Datei verfügt, können Benutzer die Zeitreihendaten in einem Plot Viewer untersuchen und als Diagramme anzeigen lassen. Benutzer können bestimmte MessageFields auswählen und wichtige Werte wie Maximum und Minimum anzeigen lassen. Zudem können sie Zeitbereiche in den Zeitreihen auswählen und hereinzoomen (Panning und Zooming).

4. Flight Comparison Page von zwei Flügen:

Diese Seite ähnelt der Detail-Page eines Fluges, ermöglicht jedoch den Vergleich von MessageFields von zwei ausgewählten Flügen. Auch im Vergleichsmodus können Benutzer zoomen und pannen. Es gibt separate Steuerelemente, um das Panning und Zooming der beiden Flüge unabhängig voneinander durchzuführen, da es wichtig ist, Zeitbereiche genau auszuwählen und zu vergleichen.

Zusätzlich zu den oben genannten Funktionen gibt es zwei Meilensteine, die nicht erreicht werden konnten:

1. Benutzerauthentifizierung und Anmeldung.

2. Ansicht zur Verwaltung von Flugdateien für einzelne Flüge (Download/Upload).

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die Hauptzielgruppe waren Mitglieder von Organisationen, die von SearchWing entwickelte Drohnen betreiben, sowie das SearchWing-Entwicklungsteam selbst. Dank dieser Anwendung sind die Flugdaten besser organisiert, und die Nutzer können sich ein vollständiges Bild davon machen, was während einer Mission passiert ist. Die App kann auch Ingenieuren helfen, Probleme in Drohnen im Voraus zu erkennen oder Risikofaktoren in bestimmten Komponenten zu finden.

Die Tatsache, dass das Projekt als Open Source veröffentlicht wird, bedeutet, dass ein großer Teil der Drohnengemeinschaft davon profitieren kann. Tatsächlich gibt es Interesse von Leuten aus dieser Gemeinschaft, den LogViewer zu hosten und für ihre eigenen Projekte zu nutzen, da er für mehrere Anwendungsfälle geeignet ist.

Was die weitere Entwicklung der App angeht, haben wir mehrere Ideen, die nach Priorität geordnet sind:

1. Benutzerauthentifizierung und Login-Funktionen

2. Unterstützung für die Verarbeitung und Speicherung von Bildern

3. Integration von Wetter-APIs, um detaillierte Wetterdaten zu liefern

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Aus Zeitmangel war es nicht möglich, die Image GUI App zu implementieren. Wir haben uns entschieden, uns auf die Flight Log Viewer App zu konzentrieren, weil es bereits einen funktionalen Prototypen der Image GUI gibt. Die wichtigsten Features der Flight Log Viewer App hingegen wurden umgesetzt bzw. wurden so aufbereitet, dass diese in den Monaten nach der Förderungsphase finalisiert werden können.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Der aktuelle Stand wird im Github mitsamt einer Doku veröffentlicht. Ebenfalls wird ein Blog Post über das Projekt auf der Internetseite https://www.hs-augsburg.de/searchwing veröffentlicht.

Gitlab: https://gitlab.com/searchwing/development/groundstation/flight-data-service

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Da wir entschieden haben, die Entwicklung der Image GUI App nicht mehr zu verfolgen, gab es keine signifikanten Anpassungen des Arbeitsaufwandes.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Unsere Arbeit, die wir als Mai Ly Le & Ricardo Martinez Garcia GbR geleistet haben, unterlag den Forderungen unseres Auftraggebers SearchWing. Auf Wunsch und in Abstimmung des Vereins SearchWing e.V. wurde entschieden, die Arbeit an der Image GUI außen vor zu lassen. Es gab neben dieser Maßnahme keine maßgeblichen Anforderungsänderungen durch SearchWing e.V. in Bezug auf die Features der Flight Log App.