



IOP Asset Tracking & VT Capstone Project

Final Presentation

May 5, 2023



Agenda

- Project Overview
 - Objective & User Stories
 - Project Management Approach & Project Calendar
- High Level Solution Architecture
- Task Area Overviews
 - Data Loading & Parsing
 - Topic Modeling & ML Search
 - Backend Services + API Layer
 - User Interface (UI)
- Final Product Live Demo
- Q&A



Project Overview

- RFP/RFI contributors currently must sift through dozens of previously submitted RFP/RFI responses for content they may want to recycle and/or build on
- This process is time-consuming and tedious, and can be fruitless, as the sections sought are often not found
- IOP has attempted to address this problem by organizing former RFP/RFI responses in folders and subfolders with clear, intuitive titles.
- While doing this helps, it does not solve the problem.



User Stories

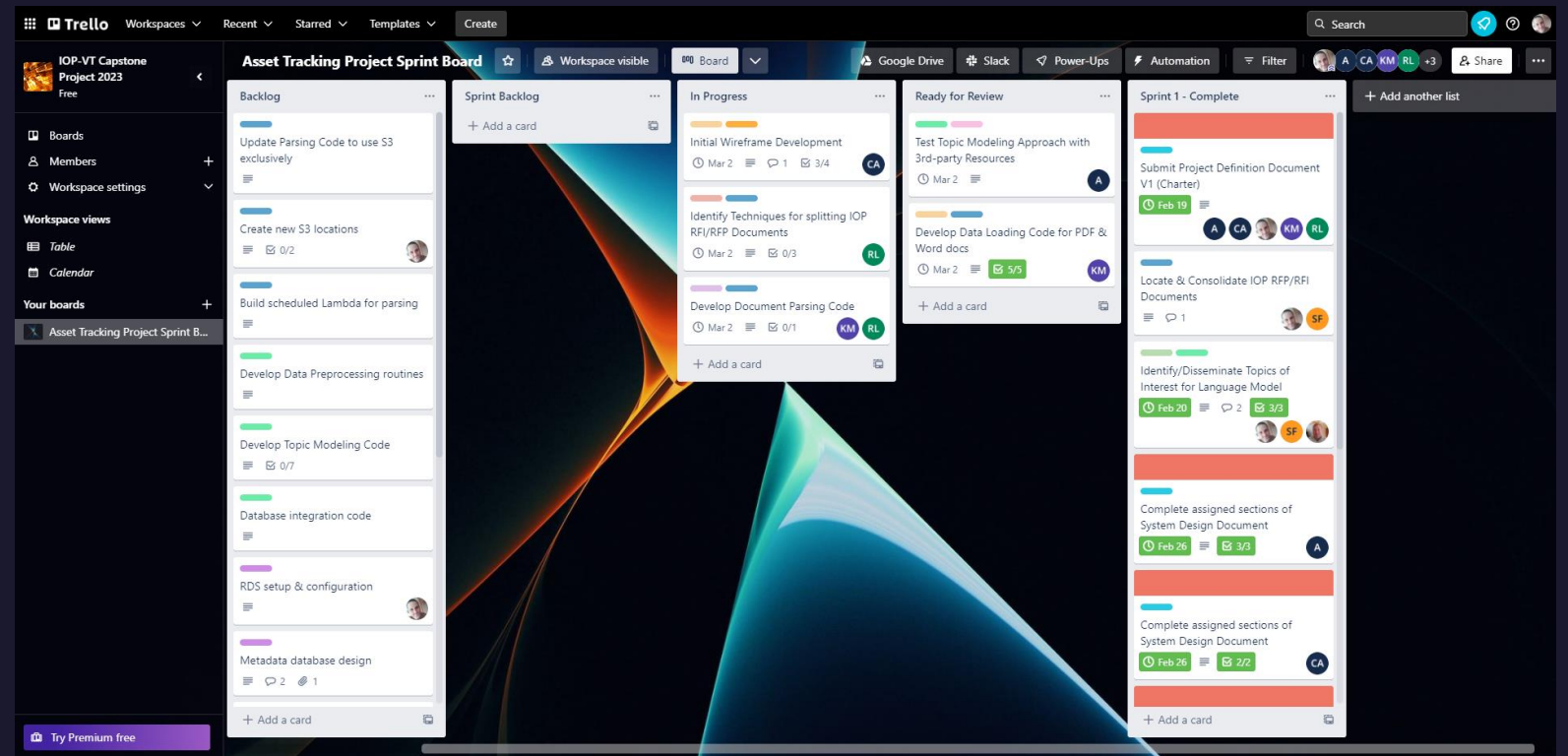
As an IOP Proposal Contributor, I want to:

- See IOP proposal docs broken down into Sections, for easy incorporation into new proposal efforts (*completed*)
- Perform basic Google-like searches across all IOP proposal doc Sections, as well as across a specified subset of docs, e.g. within a Topic (*completed*)
- Browse relevant sections of proposal docs by Topic (*completed*)
- Display matches in context and with links to the associated original document(s) (*completed*)
- Include metadata with matches, to include client, value, contract start/end date, doc type, when available (*completed*)
- Allow for filtering matches by metadata fields (*completed*)

Project Management Approach

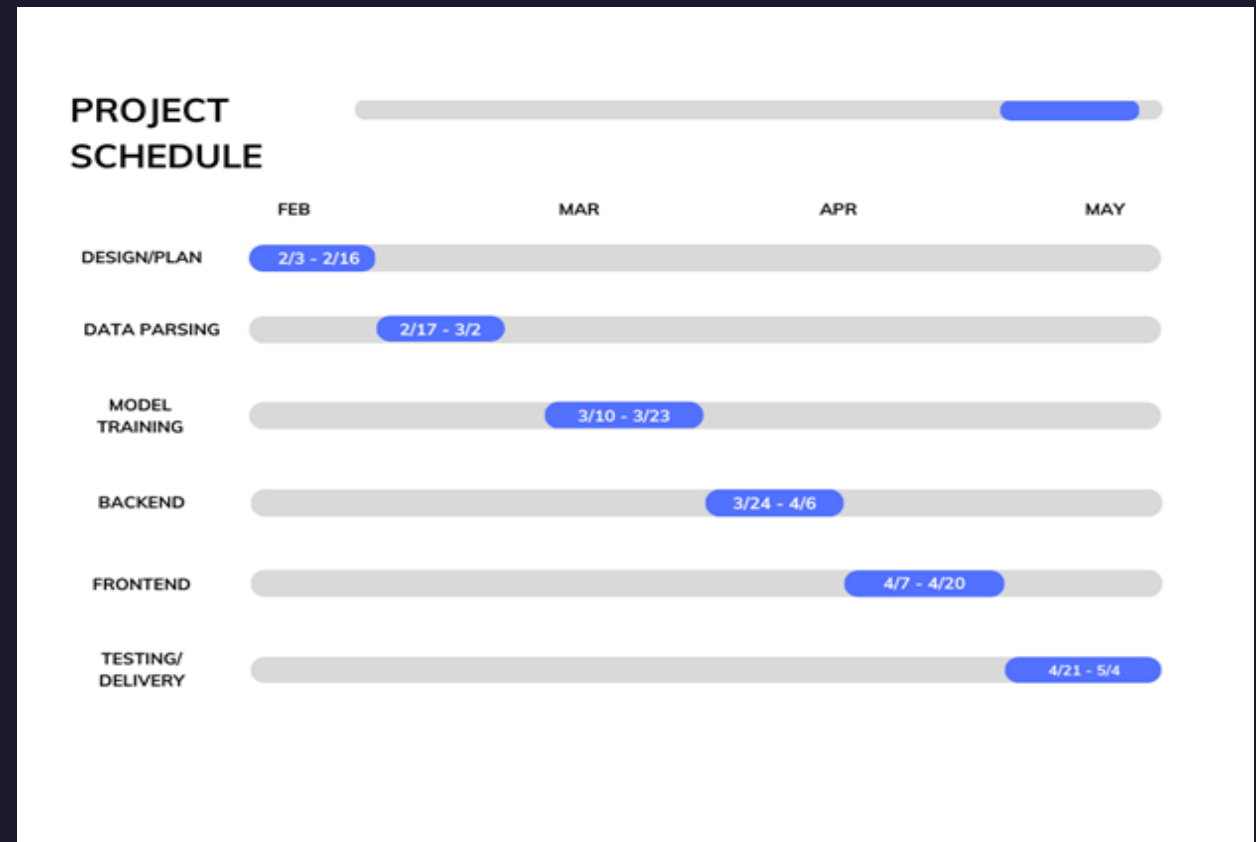


- Follow Agile SCRUM Methodology with 2-week Sprints
- Completed 6 Sprints over course of project (Sprints 0 – 5)
- Used Trello for Task Management
- Weekly Meetings w/ Client



Project Calendar (High-Level)

- Many Tasks completed in parallel, but each Sprint had an overall “Theme”
 - Sprint 0 – Design/Plan
 - Sprint 1 – Data Loading & Parsing
 - Sprint 2 – Topic Modeling/Embeddings-based Integration
 - Sprint 3 – Backend Services Development
 - Sprint 4 – Frontend (UI) Development
 - Spring 5 – Testing & Delivery
- Major Class Milestones
 - Project Definition Document Submitted – February 19
 - Project Design Document Submitted – February 25
 - Progress Report I Presentation – March 13
 - Sprint Deliverables – March 31, April 16, April 30
 - Poster Session/Final Presentation – May 1/May 5
 - Final Deliverables Due – **May 8**



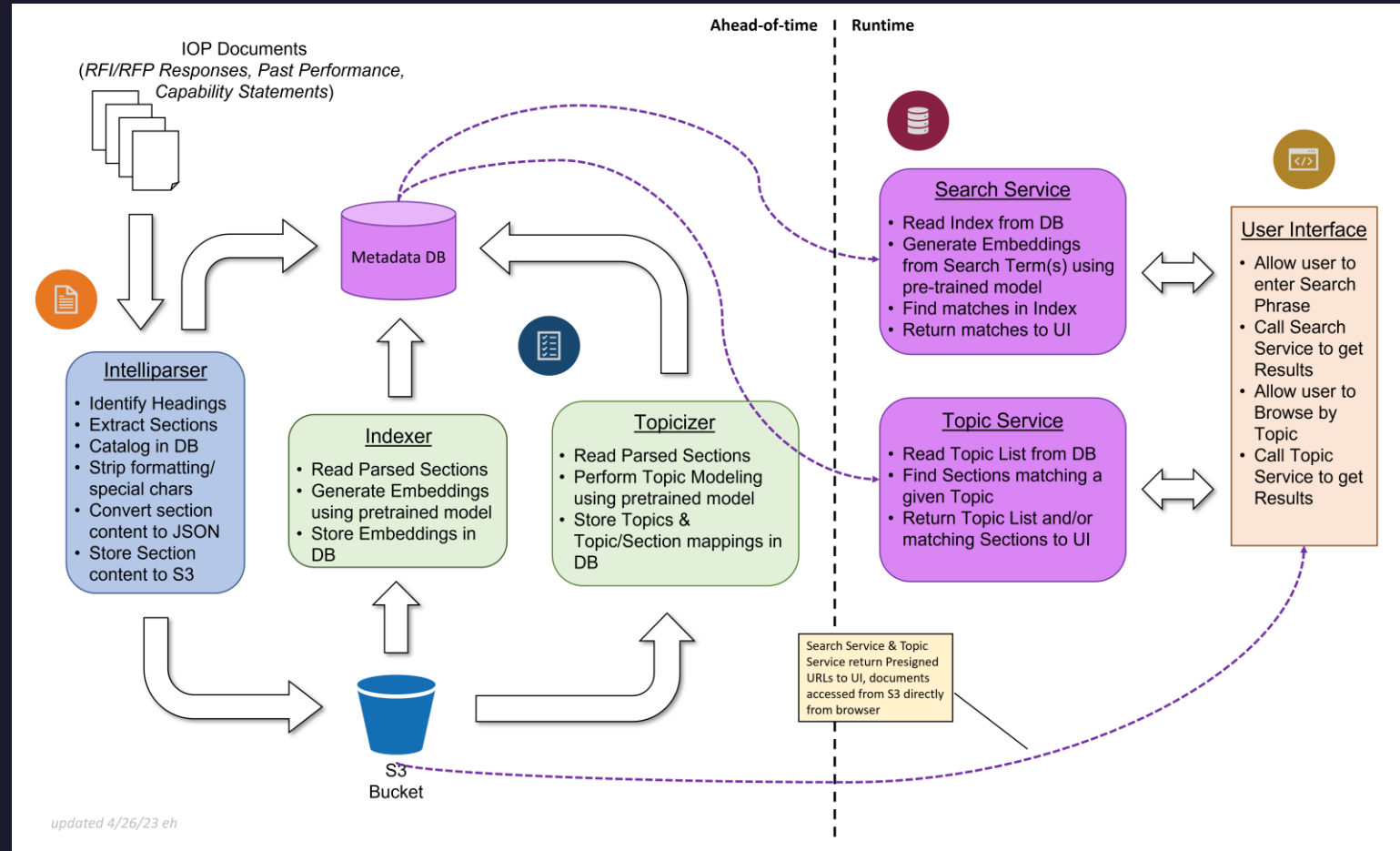
High Level Solution Architecture

- Solution has been decomposed into four major Task Areas, which loosely follow the path of data through the system:

- Document Loading & Parsing (*Ryan*)
- Topic Modeling (*Abrar*)
- Backend Service Development (*Keith*)
- Front end (UI) Development (*Camila*)

- Leveraging IOP AWS environment:

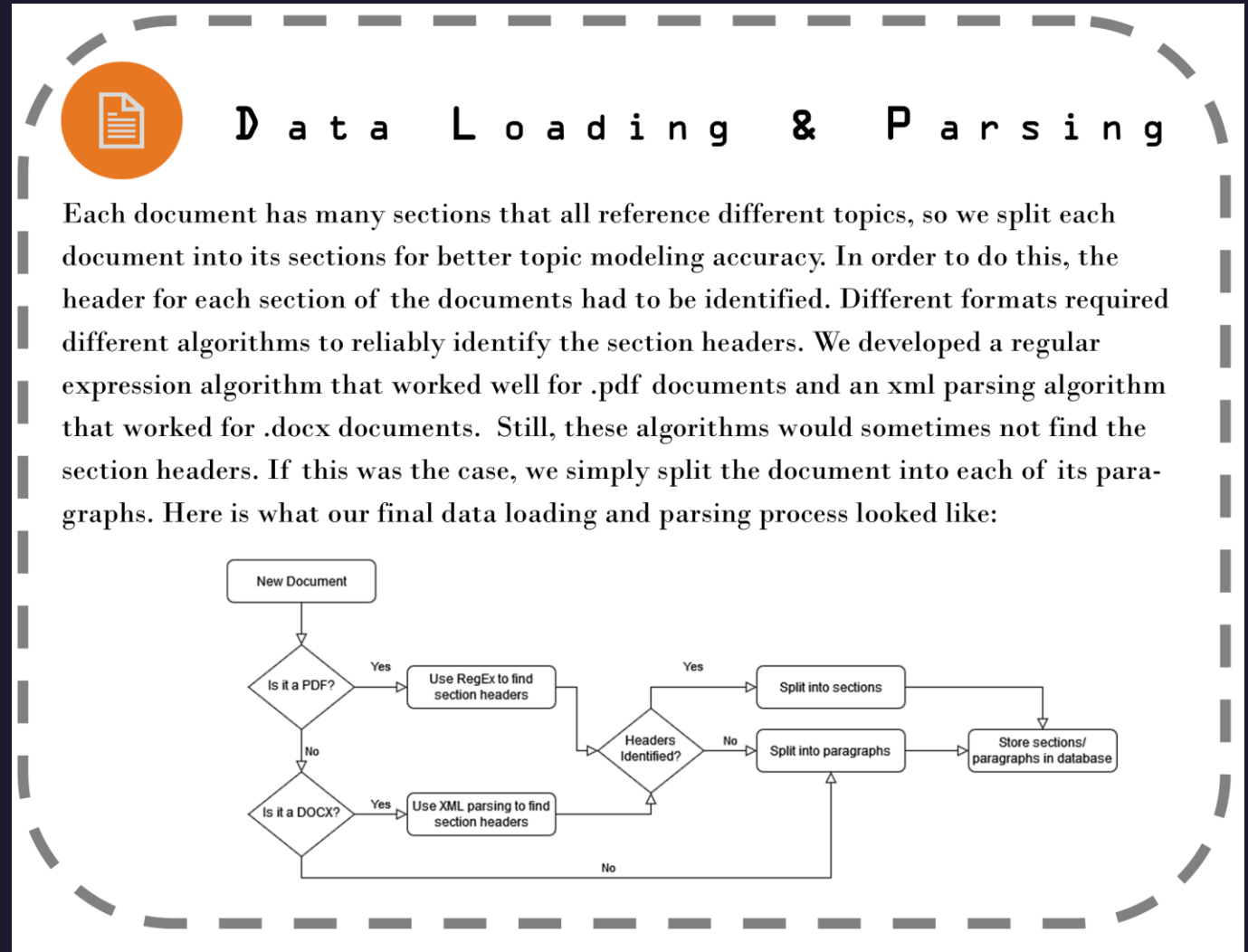
- S3 Bucket for document & parsed section storage
- RDS instance for metadata storage
- Lambda functions for backend components
- API Gateway to expose Lambdas to UI
- EFS volume for persistent storage from Lambdas



Data Loading & Parsing



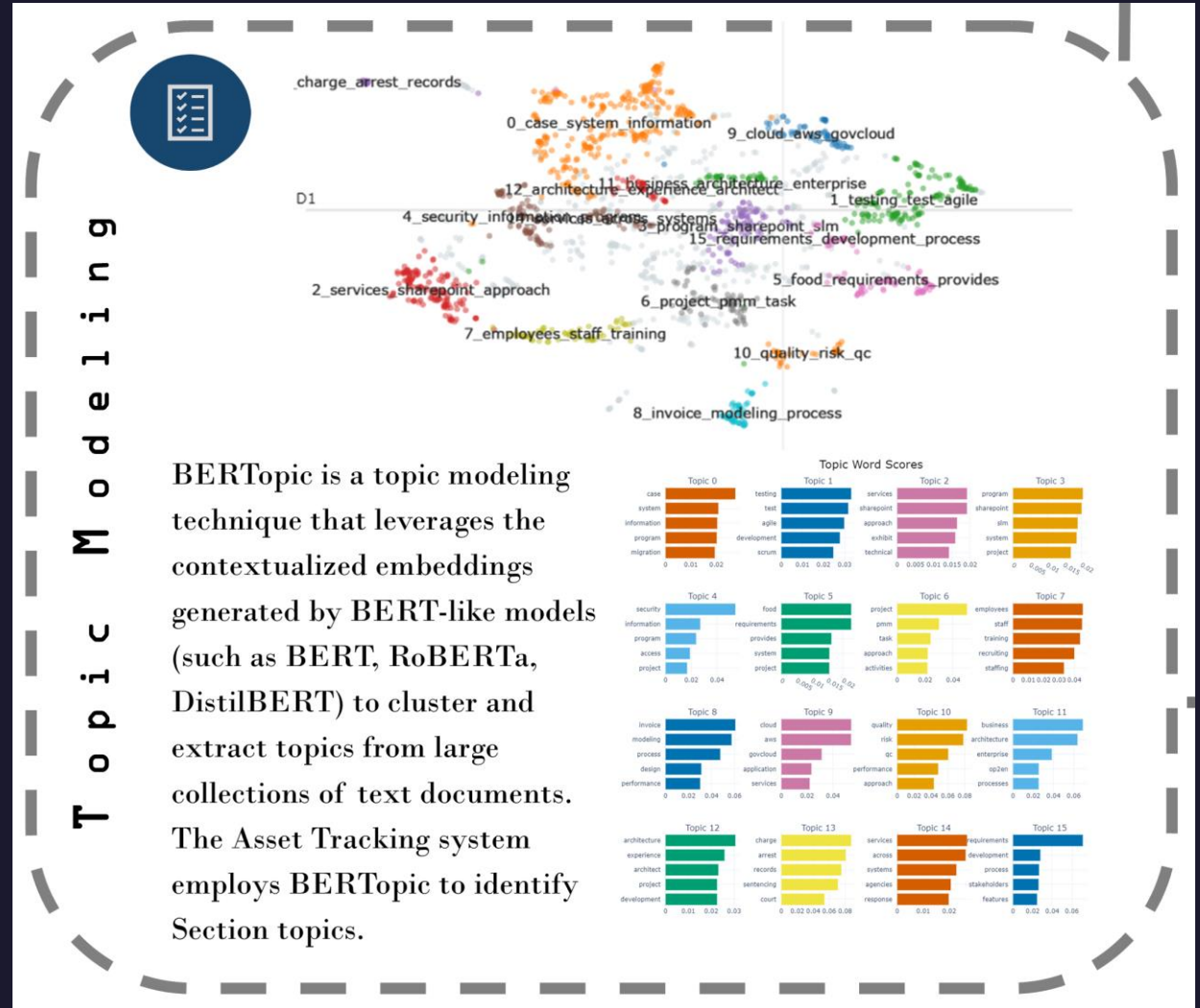
- Each document has many sections that all reference different topics, so we split each document into its sections for better topic modeling accuracy
- In order to do this, the header for each section of the documents had to be identified
- Different formats required different algorithms to reliably identify the section headers
 - We developed a regular expression algorithm that worked well for .pdf documents
 - And an XML parsing algorithm that worked for .docx documents
 - In cases where sections headers can still not be identified, we split the document into paragraphs



Topic Modeling & ML Search



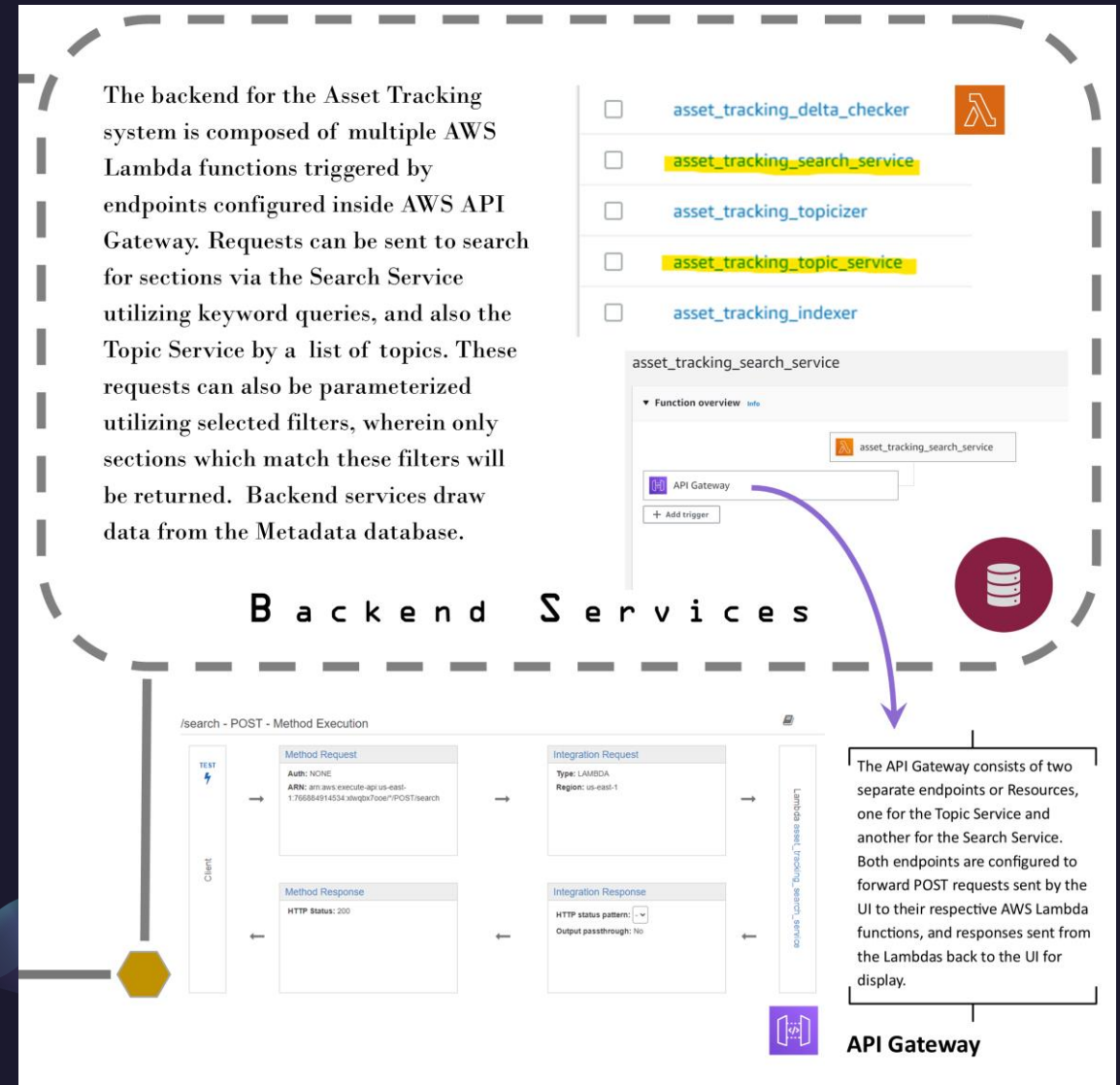
- BERTopic is a topic modeling technique that leverages the contextualized embeddings generated by BERT-like models
- Clusters and extracts topics from large collections of text documents
- The Asset Tracking system employs BERTopic to identify Section topics
- These capabilities are implemented as the “Topicizer” and “Topic Service” components
- In parallel with Topic Modeling activities, we also ported and integrated the embeddings-based search capability from IOP’s previous implementation
- These ported capabilities are implemented as the “Indexer” and “Search Service” components



Backend Services & API Layer



- Composed by multiple AWS Lambda functions
- Triggered by endpoints configured inside AWS API Gateway
- Keyword section search via. the Search Service
- Topic-based section search via. the Topic Service
- These requests can also be parameterized utilizing selected filters, wherein only sections which match these filters will be returned
- Backend services draw data from the Metadata database



User Interface (UI)



- First designed as a digital wireframe sketch using the design software Balsamiq
- The prototype was then implemented as a React web application using JavaScript, JSX, and CSS
- Most of the UI components were built using the Bootstrap and React-Bootstrap libraries
- The user can perform a filtered keyword or topic search
- Then the backend search services are leveraged to dynamically build result cards and display the most relevant results

The screenshot shows the IntegrityOne Partners web application. At the top, there's a navigation bar with the company logo. Below it, a search interface is visible with a search bar containing the text "legacy data migration" and a search button. To the left of the search bar, there are filter options for Client Organization, Document Type, and Value of Contract. The search results are displayed as a list of cards, with the first card showing details for "Result 5483" and a snippet of text: "NET applications for multiple stakeholders".

The Asset Tracking User Interface was first designed as a digital wireframe sketch using the design software Balsamiq. The prototype was then implemented as a React web application using Javascript, JSX, and CSS. Most of the UI components were built using the Bootstrap and React-Bootstrap libraries. The user can perform a filtered keyword or topic search, then the backend search services are leveraged to dynamically build result cards and display the most relevant results .

U
s
e
r
I
n
t
e
r
f
a
c
e

Solution Demo!

The screenshot displays the INTEGRITYOne Partners search interface. At the top, the logo for INTEGRITYOne Partners is visible, with the tagline 'Consulting • Technology • Innovation'. Below the logo, there is a search bar with a dropdown menu labeled 'Choose a topic...' and a search button. The search results are displayed in a list format. The first result, 'Result 5', has a distance of 0.19 and is titled 'Optional Support'. The document type is 'RFP Response'. The client organization is 'N/A', and the start and end dates of the contract are also 'N/A'. A link to the document is provided as 'Source Document'. The snippet of the document text reads: 'NET applications multiple stakeholders... IOP stands ready to provide additional optional support to ATF and the APMA in the areas of Agile and BPMS governance during the lifecycle of the project. We have extensive corporate experience in agile methodology, governance, and PMO support as well as our deep domain experience in supporting IT solutions development in the area of law enforcement investigations. We stand ready to reach back to our staff to provide ATF with additional resources as program needs require it.' Below this, 'Result 4856' is partially visible with a distance of 0.2 and a client organization of 'N/A'. On the left side, there is a 'Filters' panel with sections for 'Client Organization', 'Document Type', and 'Value of Contract'. The 'Document Type' section includes options like 'RFP Response', 'RFQ Response', 'Position Document', and 'Past Performance Only', with 'Past Performance Only' selected. The 'Value of Contract' section includes options like '<\$100k', '\$100k-\$250k', and '\$250k-\$500k', with '\$100k-\$250k' selected.

Live Demo

Thank you!

- Q&A
- Contact Info:
 - Ed Hollingsworth (eahollin@vt.edu)
 - Camila Arbaiza (camilaarb@vt.edu)
 - Ryan Lewis (lewis3rd@vt.edu)
 - Keith Myburgh (keith22@vt.edu)
 - Abrar Rhine (abrarr18@vt.edu)

