Ethical Data Science Capstone

DS496 Capstone Project Final Report

Quentin McIntosh-Giles, Hannah Hodges,

Jose Ojea, and Maria Andreina Sisco

FutureMakers, Variable Management and Reporting

Abstract: This project involved the development of a scalable dashboard designed to monitor educator engagement with FutureMakers' Sparks kits. In the absence of real-world data, a relational database was constructed using mock data generated from entity relationship diagrams. A dynamic reporting tool was then implemented using Streamlit to enable interactive exploration of key engagement metrics. The resulting prototype is fully operational and prepared to support real-time analysis as live data becomes available.

Acknowledgments

Firstly, we would like to thank Matt Barinholtz, the CEO and founder of FutureMakers for all his support, guidance, and the chance to join his team for this project. We would also like to thank Liam MacKinnon for his help on the backend of the project as well as his support and guidance. Lastly, we would like to thank Professor Drummey and Jake London for their help with the project as well.

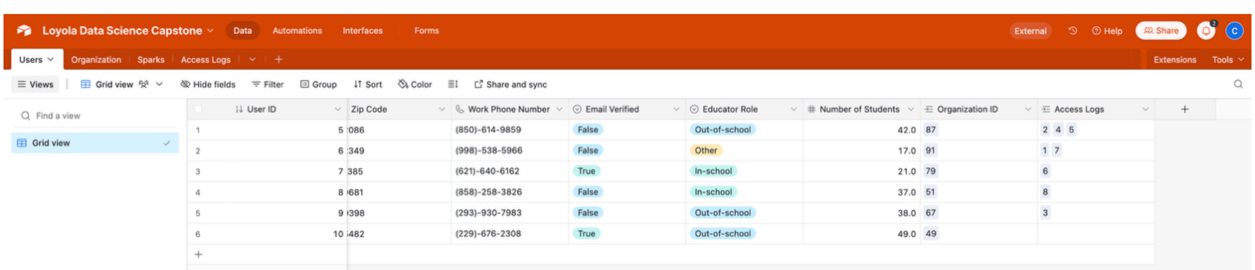
Project Description

The motivation and background of this project stems from the future of FutureMakers. Matt wanted to be able to connect with his clients with an analysis of their time usage with their sparks kit, but also wanted to be able to analyze how their clients are utilizing their products. Matt came to us wanting an MVP solution (Minimum viable product) that tracked educator engagement. Using key metrics like resource engagement, training, and usage of the platform. He also asked for an infrastructure to collect data for analysis. Once we had the scope of the project, we came up with a few research questions that we wanted to answer as well as guide the project. The first was how can we design a reliable system to track educator engagement across platforms? Next was what templates and frameworks best standardize event tracking for educator activities? And lastly, how can reporting dashboards be built to support real-time analysis at MVP launch? With these questions to guide us, we began working with Matt to accomplish this project.

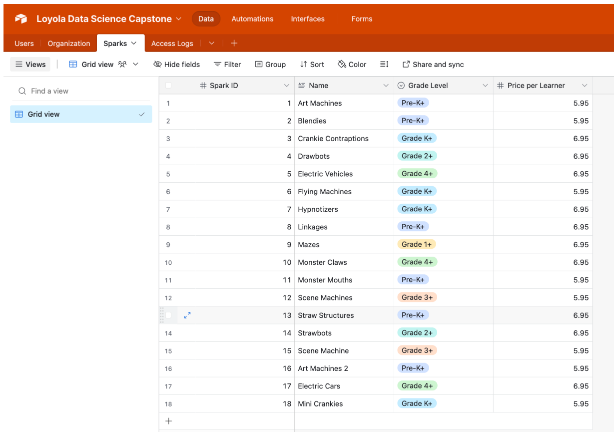
Data and Challenges

One of the most significant challenges our group encountered during this project was the lack of usable and structured data at the outset. When we first met with our client, Matt from Future Makers, we assumed that the organization already had a well-developed database in place that we could easily access, clean, and analyze. We expected to spend most of our time drawing insights and trends from existing user interaction data. However, this expectation quickly changed when we realized that the current data structure was limited and not organized in a way that would support in-depth analysis.

Rather than beginning with data analysis, we had to take a step back and essentially build a foundation for data collection and organization from the ground up. This required us to first learn Airtable, the database management platform Future Makers uses. None of us had extensive experience with Airtable prior to this project, so we dedicated time to understanding how to use the platform effectively. We explored its functionality and limitations, and then evaluated how Matt had been organizing his data so far. The existing tables included information about users, organizations, Sparks (which are the educational resources featured on the website), and some access logs. However, the data lacked consistency, completeness, and relational structure.



	User ID	Zip Code	Work Phone Number	Email Verified	Educator Role	Number of Students	Organization ID	Access Logs
1	5	086	(850)-614-9859	False	Out-of-school	42.0	87	2 4 5
2	6	349	(998)-538-5966	False	Other	17.0	91	1 7
3	7	385	(621)-640-6162	True	In-school	21.0	79	6
4	8	681	(858)-258-3826	False	In-school	37.0	51	8
5	9	398	(293)-930-7983	False	Out-of-school	38.0	67	3
6	10	482	(229)-676-2308	True	Out-of-school	49.0	49	



Loyola Data Science Capstone | Data | Automations | Interfaces | Forms

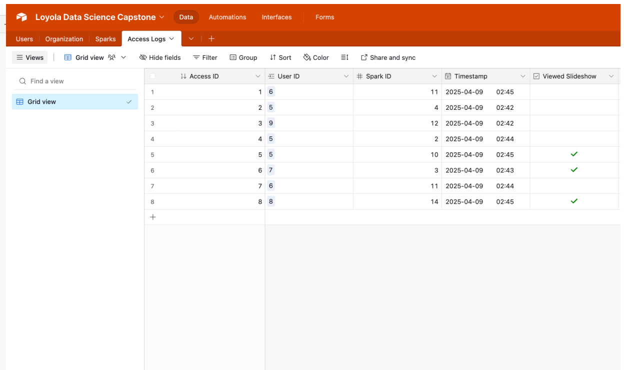
Users | Organization | Sparks | Access Logs

Views | Grid view | Hide fields | Filter | Group | Sort | Color | Share and sync

Find a view

Grid view

	Spark ID	Name	Grade Level	Price per Learner
1	1	Art Machines	Pre-K+	5.95
2	2	Blendies	Pre-K+	5.95
3	3	Crankie Contraptions	Grade K+	6.95
4	4	Drawbots	Grade 2+	6.95
5	5	Electric Vehicles	Grade 4+	6.95
6	6	Flying Machines	Grade K+	5.95
7	7	Hypnotizers	Grade K+	6.95
8	8	Linkages	Pre-K+	6.95
9	9	Mazes	Grade 1+	6.95
10	10	Monster Claws	Grade 4+	6.95
11	11	Monster Mouths	Pre-K+	5.95
12	12	Scene Machines	Grade 3+	5.95
	13	Straw Structures	Pre-K+	6.95
14	14	Strawbots	Grade 2+	6.95
15	15	Scene Machine	Grade 3+	5.95
16	16	Art Machines 2	Pre-K+	5.95
17	17	Electric Cars	Grade 4+	6.95
18	18	Mini Crankies	Grade K+	5.95



Loyola Data Science Capstone | Data | Automations | Interfaces | Forms

Users | Organization | Sparks | Access Logs

Views | Grid view | Hide fields | Filter | Group | Sort | Color | Share and sync

Find a view

Grid view

	Access ID	User ID	Spark ID	Timestamp	Viewed Slideshow
1	1	6	11	2025-04-09 02:45	
2	2	5	4	2025-04-09 02:42	
3	3	9	12	2025-04-09 02:42	
4	4	5	2	2025-04-09 02:44	
5	5	5	10	2025-04-09 02:45	✓
6	6	7	3	2025-04-09 02:43	✓
7	7	6	11	2025-04-09 02:44	
8	8	8	14	2025-04-09 02:45	✓

We spent a significant portion of our project time restructuring these data buckets in a way that made them more relational and analytical. Our group also worked closely with Matt to understand the types of questions he wanted to answer about user engagement. He was interested in knowing how educators interact with Sparks, which parts of the site receive the most engagement, and how those interactions relate to educator understanding and student outcomes. In order to answer these types of questions, we needed to think critically about how data should be captured, and which types of user actions needed to be tracked. This led us to analyze the Future Makers website to identify potential listening points, or moments where data about user behavior could be collected. These included actions like viewing a slideshow, watching a tutorial video, downloading lesson materials, or booking a support session.

To prepare the variables used in our analysis, we began by constructing an initial entity relationship diagram (ERD) to visualize the ideal data structure for Future Makers, and to facilitate understanding for Matt. This first draft (see Appendix, "Entities Diagram - First Draft") included entities such as USER, ORGANIZATION, PRODUCT, ORDER, and SATISFACTION. The intention was to treat educational resources as products, and track usage through orders and satisfaction feedback. However, after several iterations and discussions, we realized that this schema did not align well with how educators actually interact with the Future Makers platform. As a result, we made substantial adjustments. The PRODUCT entity was renamed to Spark to reflect the actual terminology used by the client, and the ORDER relationship was eliminated entirely, as it did not represent a meaningful or realistic interaction. Similarly, the SATISFACTION entity was restructured into what is now ACCESS LOG, a more accurate and flexible representation of user engagement data.

The final version of our ERD (see Appendix, "Entities Diagram - Final Draft") reflects these changes and served as the foundation for our mock database. The updated schema consists of four primary entities: USER, ORGANIZATION, SPARK, and ACCESS LOG. Each entity includes a clearly defined set of attributes, such as educator role, Spark grade level, session length, and interaction type, that we determined based on real use cases and Matt’s reporting needs. Relationships such as “works at” and “contains” were established to define how users are connected to organizations, and how their interactions with Sparks are logged. The Access Log table became the central hub for capturing key actions like viewing slideshows, booking support sessions, and using the AI Playbook Maker, all measures that the current website was already gathering or could be easily implemented. This relational model allowed us to create a normalized, scalable dataset with realistic usage patterns. From there, we wrote Python scripts to populate each table with mock data aligned to these entities and their connections, giving us a comprehensive simulated environment for testing analytical queries and visualizations.

The absence of actual user interaction data was one of the most difficult hurdles we faced. Because the infrastructure for data collection had not yet been implemented on the live website, there were no records of how educators were currently using the site. This created a major obstacle for testing and validating any analytical solutions we hoped to build. To overcome this, we generated thousands of rows of mock data that simulated realistic user interactions. We created fake users, organizations, Sparks, and detailed access logs to mimic potential site behavior. This allowed us to test the database structure, experiment with queries, and prototype the analytical tools that could be used once real data becomes available.

While the lack of real data was a challenge, it ultimately gave us the opportunity to contribute something much more foundational. Instead of simply delivering analysis, we designed and implemented a system that will allow Future Makers to capture, store, and analyze meaningful user engagement data moving forward. This system lays the groundwork for long-term insights and reporting. Our mock data served as a placeholder, but the structure we built is ready to accommodate actual usage data as the platform continues to grow. We are confident that the infrastructure we established will enable Future Makers to track educator engagement, identify trends in resource use, and support data-informed improvements to the Sparks and overall platform experience.

Analytical Methods

The technical development of the FutureMakers dashboard began with implementing a data-driven web application using Streamlit. Our primary goal was to build a tool that could visualize and analyze educator engagement with the Sparks kits. To do this, we created a system that supports interactive data uploads and dynamic visualizations based on user-provided CSV files.

The code snippet initializes this system by importing essential Python libraries—such as `pandas` for data manipulation, `plotly` and `seaborn` for rich data visualizations, and `streamlit` for building the web interface. These tools allowed us to create a clean and responsive application that educators and administrators can use without needing technical expertise.

```
#  Libraries
import streamlit as st
import pandas as pd
import plotly.express as px
import seaborn as sns
import matplotlib.pyplot as plt
from datetime import datetime
```

We configured the dashboard layout with `st.set_page_config`, giving it a professional and readable appearance. Next, we built a sidebar interface for users to upload data files relevant to engagement tracking: access logs, user data, organizational context, and Sparks resources. This setup allows the application to process real-world usage data uploaded directly by clients, making the dashboard highly adaptable.

```
# Function for generating the Account Report in Streamlit
def from_code_account_report(access_logs, users, organizations, sparks):
```

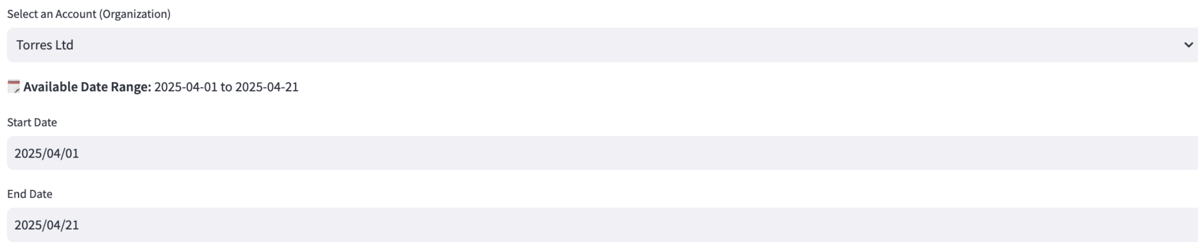
The foundational function `from_code_account_report` was designed to tie these elements together. It processes uploaded data and prepares it for reporting—paving the way for real-time insights into how educators interact with FutureMakers' offerings. By handling these operations dynamically, the dashboard empowers Matt and his team to evaluate platform adoption, identify active users, and gain a better understanding of their impact—key elements in shaping the future direction of the Sparks program.

```
# --- Streamlit App Setup ---

# Set page configuration
st.set_page_config(page_title="Spark Engagement Reports", layout="wide")

# Sidebar inputs for CSV uploads
st.sidebar.header("Upload CSV Files")
access_logs_file = st.sidebar.file_uploader("Upload access_logs.csv", type=["csv"])
users_file = st.sidebar.file_uploader("Upload users.csv", type=["csv"])
organizations_file = st.sidebar.file_uploader("Upload organizations.csv", type=["csv"])
sparks_file = st.sidebar.file_uploader("Upload sparks.csv", type=["csv"])
```

In addition to developing a functional dashboard, one of the most substantial parts of our contribution involved applying core data science concepts to generate visual insights from the mock data we built. We produced five total reports—three customer-facing and two internal. Each report generates downloadable tables and graphs based on selected parameters like organization, user, and date range.

Spark Engagement Summary

	Spark Name	User Sessions	Percent Resources Used	Timestamp
0	Scene Machine	1	14.2857	2025-04-01
1	Scene Machine	1	14.2857	2025-04-02
2	Art Machines	1	14.2857	2025-04-04
3	Mazes	1	14.2857	2025-04-04
4	Art Machines	1	14.2857	2025-04-05
5	Drawbots	1	28.5714	2025-04-05
6	Mazes	1	42.8571	2025-04-05
7	Drawbots	1	14.2857	2025-04-06
8	Blendies	1	42.8571	2025-04-10
9	Crankie Contrapti	1	28.5714	2025-04-10



This modular structure ensures the dashboard can be used flexibly across different schools, sites, and educator types, supporting both granular and summary-level analysis. The backend logic powering these reports was written in Python, leveraging pandas for data aggregation and plotly for interactivity, as stated before.

Three graphs in particular highlight the power of our approach. First, the User Activity Timeline (see Appendix: Figure 3) is generated in the Individual User Report. It provides a visual map of an educator's journey across the platform, showing the types of interactions (e.g., watching a video, booking a session) that occurred over time. The timeline format is useful for identifying patterns in how educators engage with Sparks, as well as potential drop-off points in usage. Second, the Session Length Distribution per Spark (Figure 5, from the Account Report) illustrates variation in session duration across different Sparks. While no conclusions can be drawn from the mock data, the visualization showcases how real-world usage patterns could be tracked to determine which kits are more engaging or require more time investment. Lastly, the Accesses Over Time graph (Figure 6, from the Site Report) tracks aggregate usage volume within a selected date range, allowing administrators to observe platform activity over days or weeks. Spikes in this line graph could correspond with sessions or Spark releases in a live environment.

More graphs were generated for each report, and the demo of how it all looks like live can be found in Appendix 4, Demo. All of these graphs are powered by filtering the uploaded access logs, which are parsed in real-time using functions within the dashboard. This allows the same underlying system to drive multiple reports with different user perspectives. Importantly, although these visuals are built using mock data, they fully demonstrate how real metrics will be presented and used in future decision-making. For example, the Resource Engagement Summary table (included in the report body before) lets users track resource consumption per Spark over time. Similarly, dropdown selections allow users to isolate an individual educator, a specific school, or a date range of interest. These features were designed to accommodate FutureMakers' evolving data needs, including potential integrations with Airtable or other data storage services. Overall, we applied key data science principles—ETL (extract, transform, load), data cleaning, relational modeling, and visualization—to create a functional prototype that can scale as real usage data becomes available.

Conclusions

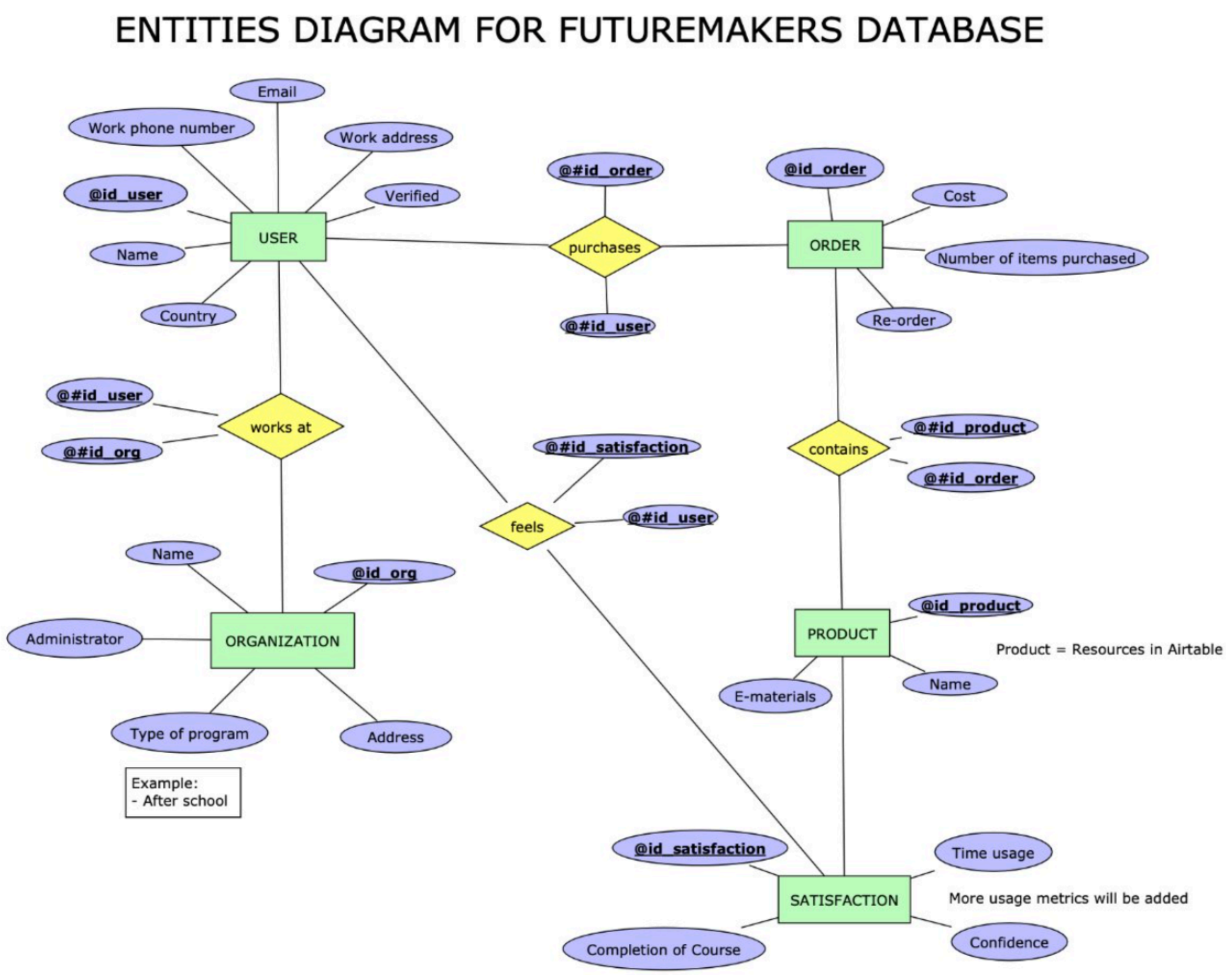
Through our analysis and dashboard development, we were able to simulate and test the reporting system that FutureMakers can use moving forward. Although we worked with mock data, the structure of our analysis allowed us to validate that the backend logic and visualizations performed as intended. For each of the five reports—Account, Site, Individual User, Sparks, and Resource Type—we confirmed that data could be dynamically filtered by date, organization, and user to generate relevant tables and graphs. These outputs responded accurately to dropdown selections and reliably reflected the conditions set by the user, proving that our ETL and filtering pipeline was functioning correctly.

More specifically, we verified that our system could compute critical engagement metrics, such as session counts per Spark, average session lengths, and the percentage of resource types accessed. We confirmed that different user roles (e.g., educators, administrators) could be associated with their respective sites and that those relationships were correctly represented in the final outputs. We also demonstrated that data from multiple entities (users, organizations, access logs) could be joined, transformed, and visualized seamlessly within a single report. While we were not able to extract insights from actual usage behavior, our results show that the system is ready to do so as soon as real data becomes available.

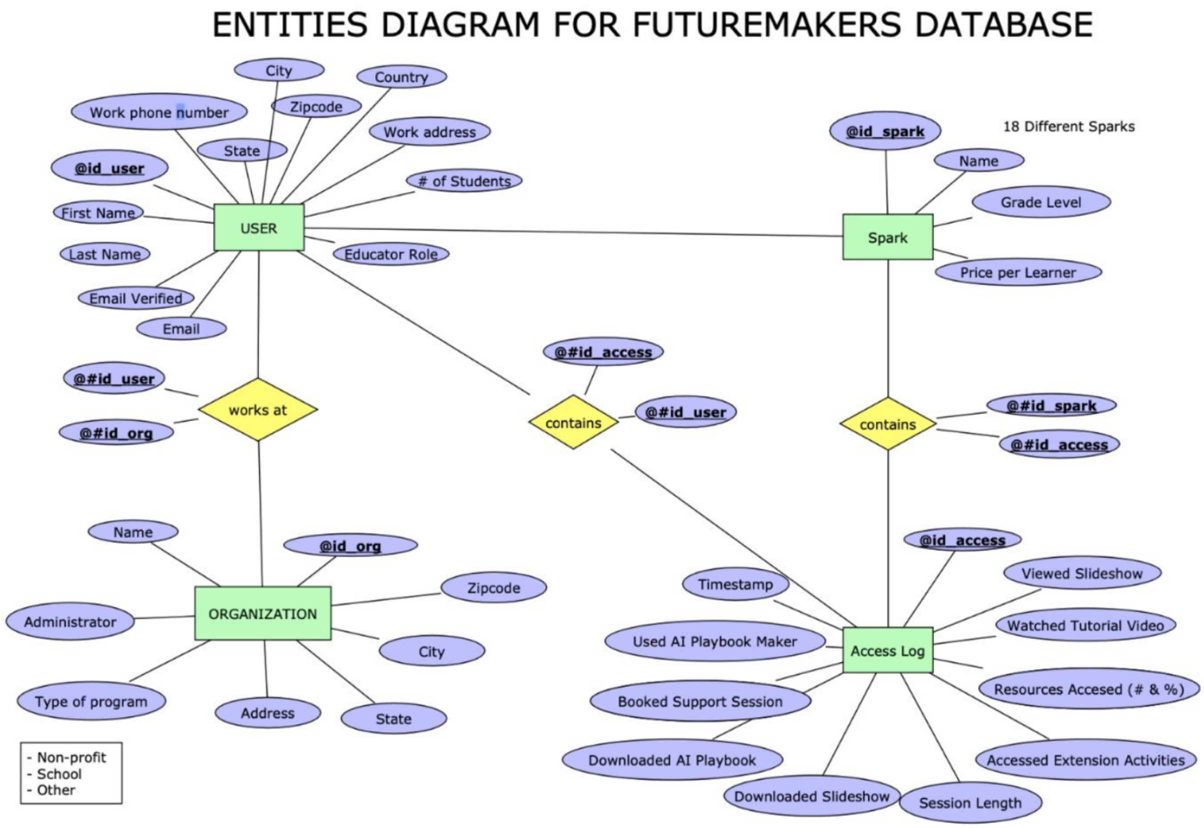
The scope of our project was not to bring to light information from the data but to formulate an interface for the client to create dashboards as easily as possibly. With the help of python and the Streamlit dataframe we can utilize the data created from the features in Airtable to pull specific columns from. Although the data itself did not reveal any insight as it was mock data, this project did reveal different metrics that we currently did not think about tracking for the client. There were several aggregated feature that can be created through Airtable which will be particularly useful when the real data starts coming in.

Appendix

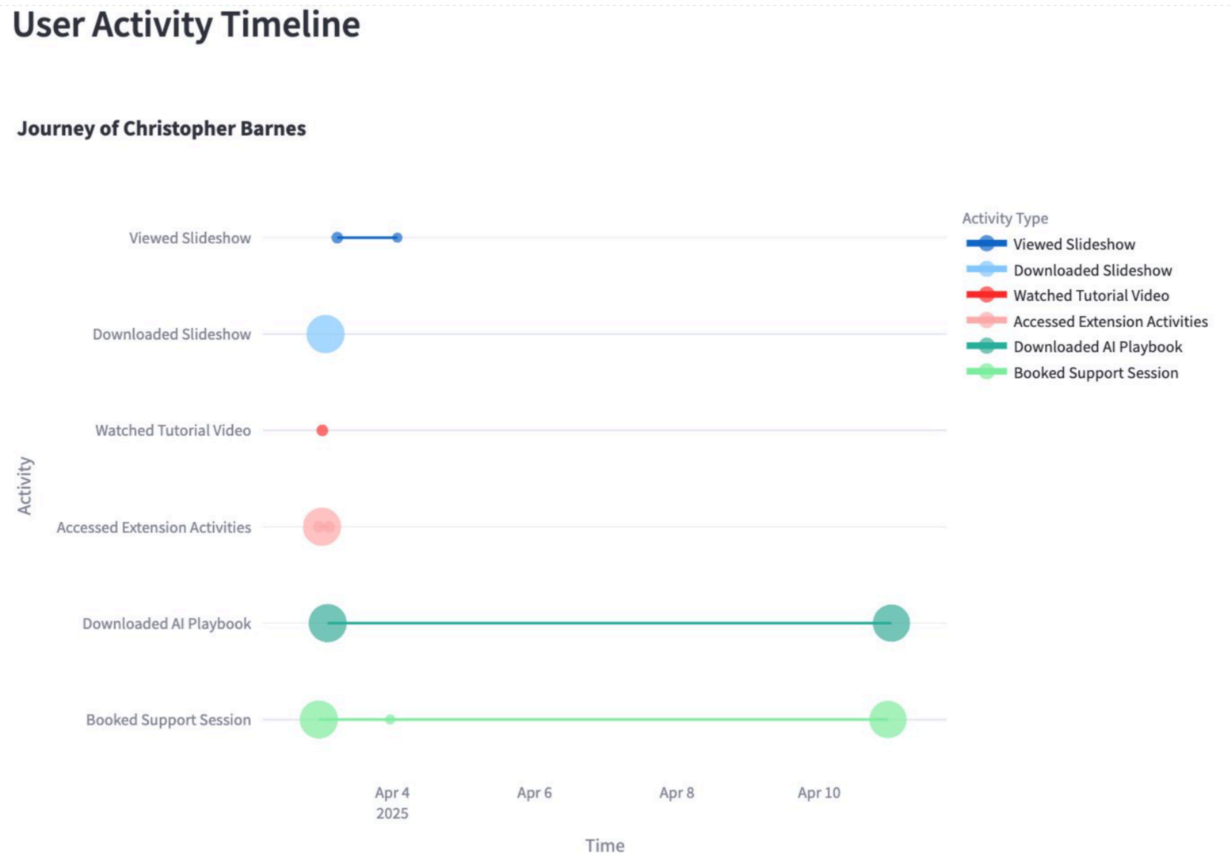
1. Entities Diagram, First Draft:



2. Entities Diagram, Final Draft:



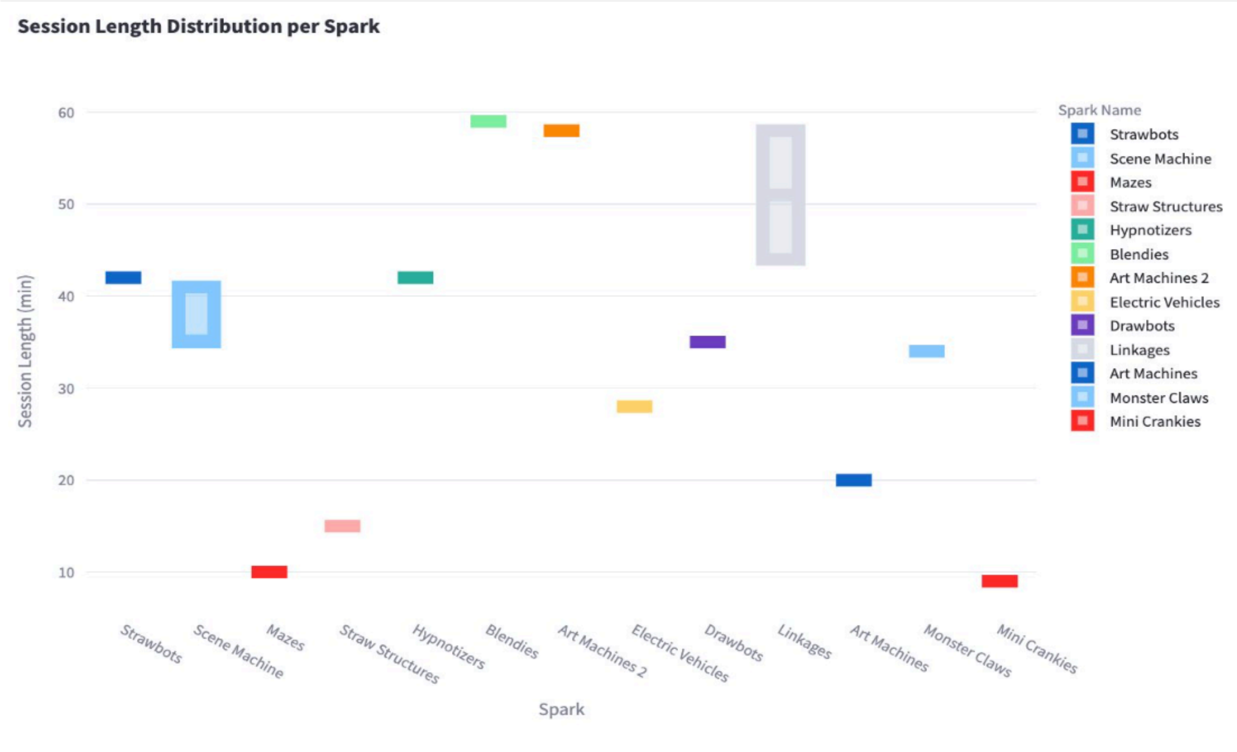
3. User Activity Timeline



4. Demo:

https://youtu.be/Av8QAjIctN0

5. Session Length Distribution per Spark



6. Accesses Over Time graph