

# INSPIRE HACKATHON SOLUTION DOCUMENTATION

Team Name : DataPulse

Team Members

Name	Email	Phone Number
Dr. Timothy Kintu	<a href="mailto:timothykintu@gmail.com">timothykintu@gmail.com</a>	256773205743
Ben Wycliff Mugalu	ben12wycliff@gmail.com	256782862788
Dr. David Jolly Muganzi	<a href="mailto:mdavidjolly@gmail.com">mdavidjolly@gmail.com</a>	256758671819
Dr Meddy Rutayisire	rutayisire24meddy@gmail.com	256780235202

## Table of Contents

<b>INSPIRE HACKATHON SOLUTION DOCUMENTATION</b>	<b>1</b>
Introduction	3
Getting Started	3
Methodology	8
Data Analysis	10
<b>Future considerations</b>	<b>12</b>

## Introduction

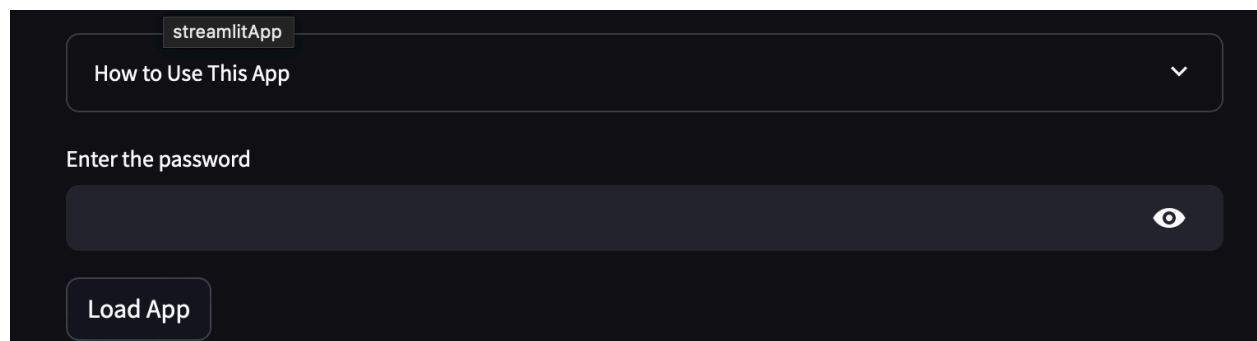
Our solution is a smart application primarily aimed at data preprocessing and similarity analysis between two datasets.

This documentation is comprised of 3 sections (Getting Started, Methodology, Data Analysis). The “Getting Started” sections explains how to set up and utilise our application. The “Methodology” section discusses the functionality, usage, and relevant information for future development or troubleshooting. The Data analysis section discloses the key insights we arrived at by exploring the synthetic data provided.

## Getting Started

**Step 1:** Click on the URL : [HERE](#)

**Step 2 :** Read instructions on how to use APP by expanding clicking on the down facing triangle sign on the right of “ How to use this App”

The screenshot shows a dark-themed web interface. At the top, there is a 'streamlitApp' header. Below it is a dropdown menu labeled 'How to Use This App' with a downward-pointing triangle icon on the right. Below the dropdown is a text input field with the placeholder text 'Enter the password'. To the right of the password field is an eye icon. At the bottom left of the interface is a button labeled 'Load App'.


**Step 3:** Input the password : “inspire\_24” , cross check by clicking on the eye icon on the right to confirm. Then click Load App

**Step 4:** Upload the data sets containing records to be matched ( Currently the App accepts two files either CSV or XLSX . One can either drag or browse )

# Record Linking Tool (Prototype)


streamlitApp

Choose the First File

 Drag and drop file here  
Limit 200MB per file • CSV, XLSX

Browse files


Choose the Second File

 Drag and drop file here  
Limit 200MB per file • CSV, XLSX



Browse files

Upon successful upload a green info box will be displayed as below


Choose the First File

 Drag and drop file here  
Limit 200MB per file • CSV, XLSX

Browse files



 synthetic\_facility\_v3.csv 238.0KB 

Choose the Second File

 Drag and drop file here  
Limit 200MB per file • CSV, XLSX

streamlitApp

Browse files

 synthetic\_hdss\_v3.csv 275.4KB 

Files uploaded successfully!

**Step 5 :** Preview the uploaded files

## Preview of synthetic\_facility\_v3

	recnr	firstname	lastname	petname	dob	sex	nationalid	patientid	visitdate
0	2	Fatuma	None	Zaina	24/08/2017 00:00	2	N_ID_5000	2,069	10/09/20
1	3	Gloria	Rashida	None	11/07/1993 00:00	2	N_ID_11861	2,079	14/12/20
2	4	Ali	Hakram	Igomu	17/05/2014 00:00	1	N_ID_11864	2,080	09/06/20
3	5	Nakalema	None	Nkwanga	27/02/2026 00:00	2	N_ID_11867	2,081	07/02/20
4	6	Asuman	Sempa	Aguti	02/03/2002 00:00	1	N_ID_11870	2,082	18/08/20

## Preview of synthetic\_hdss\_v3

	recnr	firstname	lastname	petname	dob	sex	nationalid	hdssid	hdsshhid
0	1	Zaina	Hanifa	Ula	22-09-1930 00:00	2	None	I20001	HH100001
1	2	Godfrey	Maganda	Mukama	15-07-1934 00:00	1	None	I20002	HH100002
2	3	Kasim	Ngobi	Galabuzi	03-03-1983 00:00	1	None	I20003	HH100003
3	4	Esther	None	Inara	30-07-1968 00:00	2	None	I20004	HH100004
4	5	Sumaya	Swabula	None	13-12-1930 00:00	2	None	I20005	HH100005

**Step 6 :** Select columns from each data set to be used by the algorithm to match the records ;

( Guided by the Exploratory Data Analysis above , the following columns were selected for the Facility and HDSS data set to demonstrate proof of concept )

Select columns from the synthetic\_facility\_v3:

firstname × lastname × petname × dob × sex ×

Select columns from the synthetic\_hdss\_v3:

firstname × lastname × petname × dob × sex ×

**Step 7 :** Set weights for the algorithm . The current settings default at 3 but a user can assign weights for each column for the algorithm to consider when conducting the matching of records in the uploaded datasets .

The screenshot shows a 'Set weights' interface with a dark background. It contains several sliders, each with a red line and a red dot indicating the current value. The sliders are labeled as follows: 'Weight for firstname:' (value 1), 'Weight for streamlitApp:' (value 1), 'Weight for petname:' (value 1), 'Weight for dob:' (value 1), 'Weight for sex:' (value 1), 'Weight for firstname:' (value 1), and 'Weight for lastname:' (value 1). Each slider has a '3' at the right end, indicating the maximum weight. A small 'streamlitApp' label is visible next to the second slider.

**Step 8 :** Select Number of records to match . A user can decide to either match All records or choose a number guided by the total number of records as indicated in the text output above the prompt

The screenshot shows a form with a dark background. It contains the following text: 'Total records in synthetic\_facility\_v3.csv: 2902', 'Total records in synthetic\_hdss\_v3.csv: 4115', and 'How many records do you want to compare? Enter a number or "All" for all records.' Below the text is a text input field with the value 'All'.

**Step 9 :** Select a Cut off Score of similarity for the algorithm to use for the output . 100 % reflects a perfect match across the selected columns . The app Defaults at 90%

Set Cutoff Score

Choose a Cutoff Score - Recommended is 90

0

90

100

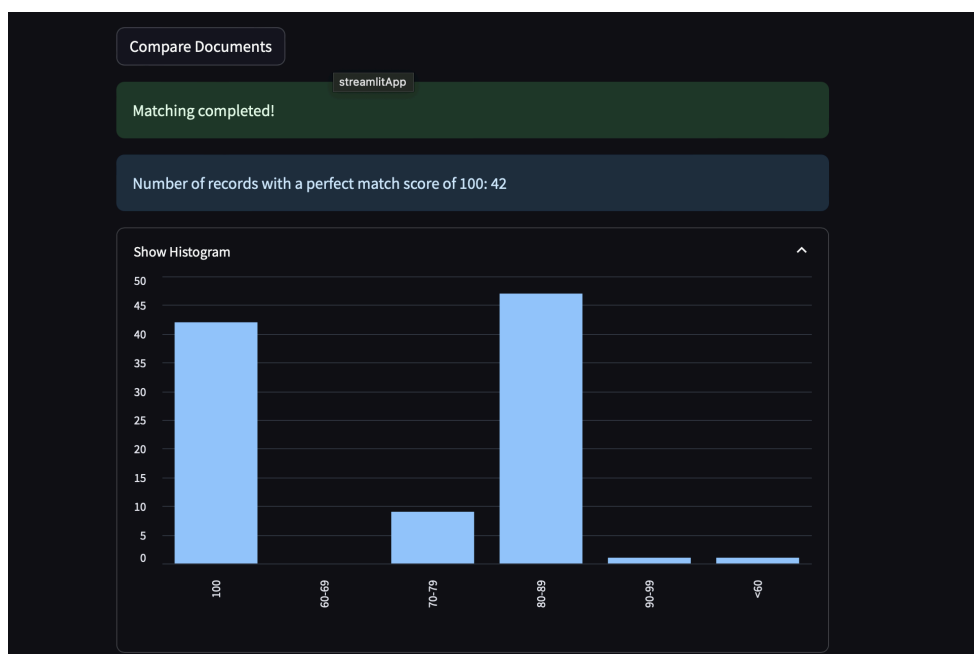
The selected cutoff score is: 90

Step 10 : Start the comparison by clicking on button “ Compare Documents” .

Compare Documents

Performing Matching...

**Step 11:** Review results of the matching . A green info box will be displayed signaling that the matching of the selected number of records is complete . A blue info box will show how many records had a perfect match for the selected columns , one can expand to see a histogram showing distribution of the matching scores.



**Step 12 :** Display and or Download the results. The user can expand the “Display Results” box to preview the results ( records from both data sets with corresponding scores arranged in Descending order) . One can download a CSV of the results of matching ( Possible matches and the respective scores) by clicking on the “Download Matching records as CSV” Button for further exploration.

Display Results

	synthetic_facility_v3.csv_firstname	synthetic_facility_v3.csv_lastname	synthetic_facility_v3.csv_pet
1	gloria	rashida	None
79	sarah	brenda	None
51	favour	mariam	None
53	justine	patience	None
62	nakiyemba	nakaziba	haniya
71	mukama	alamazani	munyagwa
73	mutesi	phiona	None
75	aisa	namuwaya	xara
80	wambi	mugabi	kaggwa
35	isiko	akisamu	mwesigwa

Download matching records as CSV

Approach Used by This App

For more information on how the algorithm underneath the APP works click on the “Approach Used by This App” button .



## Methodology

In order to realize the logic for our application, we went through the following steps:

### 1. Dataset Loading

The `load_data(file)` function takes a file object as input and loads the data from the file. Users can upload datasets in CSV or XLSX format using file uploaders provided in the application interface. The tool dynamically detects the file format and loads the data accordingly using Pandas' `read_csv` or `read_excel`

2. Preprocessing: The code includes preprocessing functions to clean and standardize the data before matching. This includes converting text to lowercase, removing special characters, and handling missing values.

The `preprocess_string(x)` function preprocesses a single string by applying lowercase conversion, whitespace trimming, and removal of special characters.

It takes a string `x` as input and performs the specified cleaning operations. The cleaned string is then returned.

### 3. Similar Column Identification:

This function performs fuzzy matching between records from two DataFrames.

It takes two DataFrames (`df1` and `df2`) along with their corresponding column lists (`columns1` and `columns2`). Additionally, it accepts lists of weights (`weights1` and `weights2`) to prioritize the importance of columns in the matching process.

The `num_records` parameter specifies the number of records to compare, with 'All' indicating comparison of all records. The function iterates over each record in `df1` and finds the best match in `df2` based on fuzzy string similarity scores. It calculates a weighted average score for each match, considering the specified weights for columns. The matching results are stored along with the score and later sorted based on the score in descending order. Score bins are computed to summarize the distribution of matching scores. Finally, the function returns the sorted matching results DataFrame, along with the score bins for visualization.

## Data Analysis

As part of our exploration of the provided Synthetic dataset, we arrived at the following insights.

### 1. HDSS dataset

- a. All values in the national ID column in hdss dataset are missing. But this column can be important as a unique identifier if all values are available.
- b. Lastname and petname have some missing values, but firstname, dob and sex do not have any missing values
- c. No records share firstname, lastname, petname, sex and dob.
- d. No records share firstname, petname, sex and dob
- e. No records share firstname, lastname, sex and dob
- f. No records share household identifier. So every person in the data set is from a unique household.

### 2. Health facility dataset

- a. Last name has 67 missing values. There are 264 records with shared firstname, petname, dob and sex. I.e. it is possible that someone visited the health facility more than once, and on one of these occasions the lastname was not captured.
- b. Petname has 391 missing values. There are 271 records with shared firstname, lastname, dob and sex
- c. There are 259 records in the health facility dataset that share firstname, lastname, petname and dob. It is possible that these are true duplicates
- d. 1805 persons are not duplicated, 284 persons are duplicated twice, 25 persons are duplicated thrice, 1 person is duplicated four times. Duplication here may indicate that the person visited the facility more than once
- e. There are dobs after 2024, e.g 2026 for person 5, how do we go about calculating age for these people? If we consider manipulating the dob.

### 3. There seem to be 870 records shared across the datasets (if all duplicates with a shared firstname, lastname, petname, dob and sex are removed from the health facility dataset).

## Future considerations

- Performance Optimization:
  - Explore parallel processing or multi-threading for the fuzzy matching process to speed up comparisons, especially when dealing with a large number of records.
- User Experience Improvements:
  - Introduce interactive data visualization tools for users to explore the matching results more intuitively (e.g., interactive charts to drill down into match quality).
  - Provide a more detailed progress indicator during the matching process, including estimated time to completion.
- Advanced Data Preprocessing Features:
  - Add options for more sophisticated data cleaning techniques, such as stemming, lemmatization, and the removal of stopwords, to improve the quality of text comparison.
  - Implement a feature guided by Machine Learning to automatically detect and suggest which columns to compare based on their similarity or data type.
- Scalability and Deployment:
  - Containerizing the application with Docker to simplify deployment and ensure consistency across different environments.
- Security Enhancements:
  - Enhance data privacy and security measures ( App log in ) , especially for sensitive information, by implementing encryption for data in transit and at rest.
  - Introduce more robust authentication mechanisms and role-based access control to ensure that only authorized users can access the application and perform certain actions.
- Customization and Flexibility:
  - Allow users to customize the algorithm parameters for fuzzy matching to fine-tune the balance between accuracy and performance according to their specific needs.

- Provide options for users to define custom score bins or matching criteria that better fit their particular use case.
- Documentation and Support:
  - Improve the documentation, including tutorials, examples, and best practices, to help users get the most out of the application.
- Expand File Format Support:
  - Extend support for additional file formats beyond CSV and XLSX, such as JSON, XML, or even direct connections to SQL databases, to accommodate a wider range of data sources.
- Threshold Adjustment: Users may need to experiment with different similarity thresholds to optimize the matching process based on the characteristics of their datasets and specific matching requirements.
- Enhanced Matching Algorithms: Future development efforts could focus on improving the matching algorithms to handle more complex data scenarios and provide higher accuracy in matching results.
- Error Handling: Future work can include robust error-handling mechanisms to handle unexpected inputs or data inconsistencies gracefully. This would enhance the reliability and usability of the application.
- User Feedback: Collecting user feedback and incorporating user suggestions for continuous improvement and refinement of the tool's functionality.