DLP - Product Requirements Doc

A Product Requirements doc focuses on the users' idea of what the finished product should do.

✏️ Intro & Goals

The Data Lake Project is an ETL exercise on different forms of Data Lake data. Using Python Pandas, Jupyter Notebook, Seaborn we clean, unite, and explore data from CSV, JSON, to SQLite, here are some KPIs we would like in our Data Warehouse:

- ✓ View Extracted and Cleaned any datasets provided from the Data Lake
- ✓ Clear understanding of data without extra joins or merges from Data Lake
- ✓ Have unnecessary columns or rows of data removed that do not add value in analysis
- ✓ Ability to query for meaningful information on all datasets

👫 Who's it For?

The Data Lake ETL project is meant for Data Analysts and Data Scientists who are looking to create information and findings from a polished Data Warehouse.

👓 Why are we Building it?

To practice manipulating and altering data in a real-life scenario. The variation in data topics shows the range of the technologies used for cleaning and presentation.

🎯 Expected Results

Users to find data in Warehouse

- Easy to use
- Easy to query
- Easy to draw conclusions from

🔧 Product Requirements

- Columns of data reflect exactly what our data model should contain (no more and no less)

- Contain models that are hand-selected to be useful in relation to the vast data we have in our Data Lake
- Present Schemas and Models that help guide user direction in different Lake connections

🎨 User interaction and design

Example of Joins and Filtering out Columns for our Warehouse consumption:

```
Match = pd.merge(Match, Team, left_on='Team_1', right_on='Team_Id', how='left')
Match.rename(columns={'Team_Name': 'Team 1'}, inplace=True)

Match = pd.merge(Match, Team, left_on='Team_2', right_on='Team_Id', how='left')
Match.rename(columns={'Team_Name': 'Team 2'}, inplace=True)

Match = pd.merge(Match, Team, left_on='Toss_Winner', right_on='Team_Id', how='left')
Match.rename(columns={'Team_Name': 'Toss Winner'}, inplace=True)
Match = Match.drop(columns=['Team_Id_x', 'Team_Id_y', 'Team_Id'])

Match = pd.merge(Match, Team, left_on='Toss_Decide', right_on='Team_Id', how = 'left')
Match.rename(columns={'Team_Name': 'Toss Decider'}, inplace=True)

Match = pd.merge(Match, Team, left_on='Match_Winner', right_on='Team_Id', how = 'left')
Match.rename(columns={'Team_Name': 'Match Winner'}, inplace=True)

Match = pd.merge(Match, Venue, left_on='Venue_Id', right_on='Venue_Id', how='left')
Match = pd.merge(Match, Win_By, left_on='Win_Type', right_on='Win_Id', how='left')
Match = pd.merge(Match, Outcome, left_on='Outcome_type', right_on='Outcome_Id', how = 'left')
Match = pd.merge(Match, Player, left_on='Man_of_the_Match', right_on='Player_Id', how='left')
```

```
Match = Match[['Match_Id',
               'Team 1',
               'Team 2',
               'Match_Date',
               'Venue_Name',
               'Toss Winner',
               'Toss Decider',
               'Win_Type_y',
               'Win_Margin',
               'Match Winner',
               'Player_Name'
            ]]

Match = Match.rename(columns={
    'Match_Id': 'Id',
    'Match_Date': 'Date',
    'Venue_Name': 'Venue',
    'Win_Type_y': 'Win Type',
    'Win_Margin': 'Win Margin',
    'Match Winner': 'Winner',
    'Player_Name': 'MVP'
})
```

```
Match.head()
```

	Id	Team 1	Team 2	Date	Venue	Toss Winner	Toss Decider	Win Type	Win Margin	Winner	MVP
0	335987	Royal Challengers Bangalore	Kolkata Knight Riders	2008-04-18 00:00:00	M Chinnaswamy Stadium	Royal Challengers Bangalore	Kolkata Knight Riders	runs	140.0	Kolkata Knight Riders	BB McCullum
1	335988	Kings XI Punjab	Chennai Super Kings	2008-04-19 00:00:00	Punjab Cricket Association Stadium, Mohali	Chennai Super Kings	Royal Challengers Bangalore	runs	33.0	Chennai Super Kings	MEK Hussey
2	335989	Delhi Daredevils	Rajasthan Royals	2008-04-19 00:00:00	Feroz Shah Kotla	Rajasthan Royals	Royal Challengers Bangalore	wickets	9.0	Delhi Daredevils	MF Maharoof
3	335990	Mumbai Indians	Royal Challengers Bangalore	2008-04-20 00:00:00	Wankhede Stadium	Mumbai Indians	Royal Challengers Bangalore	wickets	5.0	Royal Challengers Bangalore	MV Boucher
4	335991	Kolkata Knight Riders	Deccan Chargers	2008-04-20 00:00:00	Eden Gardens	Deccan Chargers	Royal Challengers Bangalore	wickets	5.0	Kolkata Knight Riders	DJ Hussey

🙋‍♂️ Open questions

Where would we like to store our Data Warehouse?

What other forms of data would we expect to perform ETL on?

What other technologies can we utilize to enhance or optimize the ETL process of a Lake?

Are we following industry standards in formatting and design patterns?