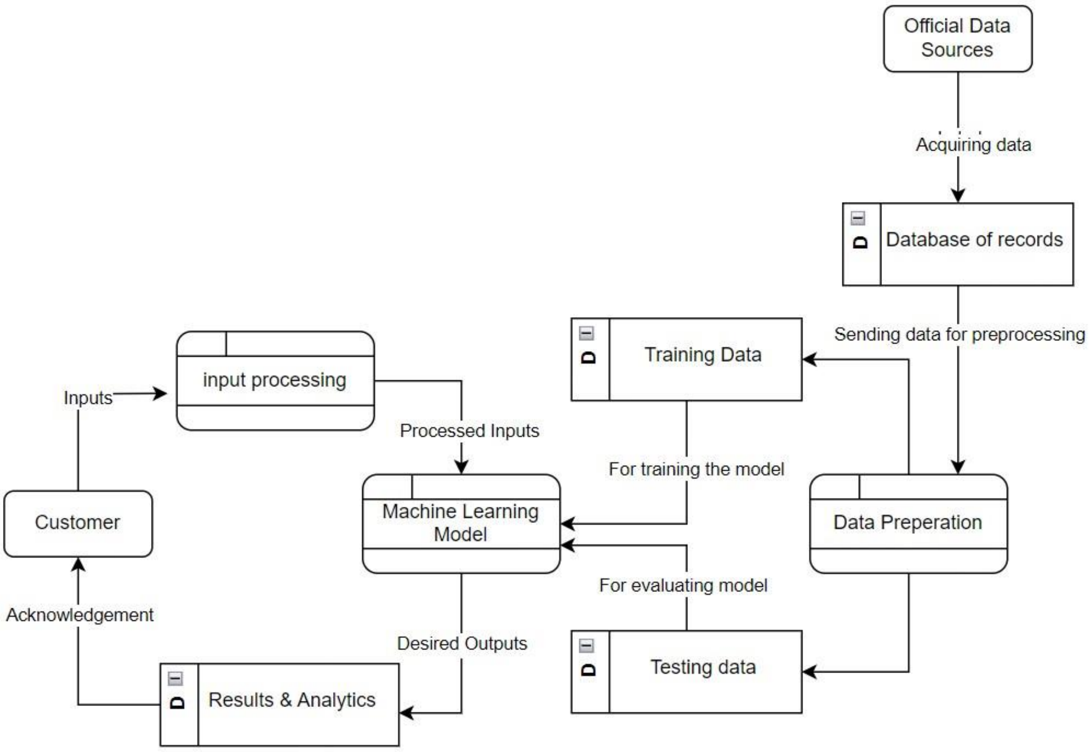
Project Design Phase-II
Data Flow Diagram & User Stories

Date	11 November 2023
Team ID	Team-591679
Project Name	T20 TOTALITARIAN: MASTERING SCORE PREDICTIONS
Maximum Marks	4 Marks

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



In the context of "T20 Totalitarian: Mastering Score Predictions Using Machine Learning," the following Data Flow Diagram (DFD) illustrates the flow of data within the system, including data acquisition, preprocessing, model training, user interaction, and result output:

Explanation of the Data Flow Diagram (DFD):

1. External Entities:

- Official Data Source: Represents the source from which historical cricket match data is obtained. It acquires and stores data into the databases of records.

- Customer/User: Represents the end-user or customer who inputs data and interacts with the system to receive predictions.

2. **Processes:**

- Data Acquisition: The official data source provides historical match data, including player statistics, match details, venue information, weather conditions, etc., which is stored in the system's databases.

- Preprocessing: Raw data is sent for preprocessing, where it undergoes cleaning, transformation, and feature engineering to prepare it for machine learning model input. pre-processed data is split into training and testing datasets.

- Model Training and Evaluation: The training data is used to train the machine learning model, which is then evaluated using the testing data to assess its accuracy and performance.

- User Input Processing: User inputs or data provided by customers are pre-processed before being fed into the machine learning model.

- Prediction Generation: The machine learning model processes the pre-processed input data to generate predictions for T20 match scores based on the trained model.

- Result Output: The predicted results are delivered to the customer or end-user, providing them with the desired output in the form of score predictions.

3. Data Stores:

- Databases of Records: Store the acquired historical match data from the official data source. This data is used for preprocessing and model training.

4. Data Flows:

- Data Flow from Official Data Source: Historical match data flows from the official data source to the databases of records for storage and further processing.

- Pre-processed Data Flow: Raw data is pre-processed and split into training and testing datasets for machine learning model input.

- Customer Input Data Flow: Inputs from customers are processed and sent to the machine learning model.

- Prediction Output Data Flow: Predicted results from the model are sent back to the customers.

The DFD outlines the flow of data within the system, starting from data acquisition, preprocessing, model training, user interaction, and the output of predictions, showcasing how information flows and transforms within "T20 Totalitarian: Mastering Score Predictions Using Machine Learning."

User Stories

list of the user stories.

User Type	Functional Requirement (Epic)	User Story Number	User Story/ Task	Acceptance Criteria	Priority	Release
Customer	Dashboard	USN -1	As a user, I can get a good interface on the dashboard to give my inputs	I can get a good dashboard	Low	Sprint – 3
Sports Analyst	Predictions	USN -2	As a Sports Analyst, I want results that are more than 90% reliable	Reliable results	High	Sprint – 2
Customer	Inputs	USN -3	As a user, I can give current situation through inputs	Inputs for considering the current status	Low	Sprint – 3
Admin	Machine Learning model	USN - 4	As an Admin, I want a pretrained machine-learning model for the implementation	Pre-trained machine learning model	High	Sprint – 1
Admin	Data processing	USN -5	As an Admin, I want data to get pre-processed before sending to Machine Learning model	Processing of data	High	Sprint – 1

Here is a more detailed explanation of the user stories for the project "T20 Totalitarian: Mastering Score Predictions Using Machine Learning":

User Story 1: Dashboard

As a user, I can get a good interface on the dashboard to give my inputs.

Acceptance Criteria:

- The dashboard should be easy to use and navigate.
- The dashboard should provide clear and concise instructions for providing inputs.
- The dashboard should be able to capture all relevant inputs from the user.

Priority: Low

Release: Sprint 3

User Story 2: Predictions

As a Sports Analyst, I want results that are more than 90% reliable.

Acceptance Criteria:

- The machine learning model should be able to predict the score of a T20 cricket match with an accuracy of at least 90%.

- The machine learning model should be able to generalize to new data and maintain its accuracy.
- The machine learning model should be able to identify the key factors that influence the outcome of a T20 cricket match.

Priority: High

Release: Sprint 2

User Story 3: Inputs

As a user, I can give current situation through inputs.

Acceptance Criteria:

- The system should allow the user to input all relevant information about the current situation of the match.
- The system should validate the user's inputs to ensure that they are accurate and complete.
- The system should store the user's inputs in a secure and accessible manner.

Priority: Low

Release: Sprint 3

User Story 4: Machine Learning Model

As an Admin, I want a pretrained machine-learning model for the implementation.

Acceptance Criteria:

- The machine learning model should be trained on a large dataset of historical T20 cricket match data.
- The machine learning model should be carefully tuned to achieve the highest possible accuracy.
- The machine learning model should be documented and made available to the development team.

Priority: High

Release: Sprint 1

User Story 5: Data Processing

As an Admin, I want data to get pre-processed before sending to Machine Learning model.

Acceptance Criteria:

- The data should be cleaned and formatted to remove any errors or inconsistencies.
- The data should be transformed into a format that is compatible with the machine learning model.
- The data should be stored in a secure and accessible manner.

Priority: High

Release: Sprint 1

Conclusion

The project "T20 Totalitarian: Mastering Score Predictions Using Machine Learning" has concluded the user stories. If these user stories are implemented successfully, a strong and dependable machine learning system that can correctly predict the score of Twenty20 cricket matches will be created. Both sports experts and bettors as well as fans will find this technique to be a useful resource. The interface will be simple to use and offer brief, precise directions for entering data. Additionally, the system will be able to record all pertinent user inputs and validate them to guarantee accuracy and completeness. The user's inputs will be safely and easily stored by the system. A sizable dataset of previous T20 cricket matches will be used to train the algorithm, data and meticulously adjusted to attain optimal precision. The development team will have access to the system's documentation. To make the data suitable for the machine learning model, it will be cleaned, formatted, and checked for mistakes or inconsistencies.