Date to enter includes

|  |
| --- |
| Player given name: |
| Opponent given name: |
| Match date: |
| Match number: |
| Game number: |
| Initial server: |

The user enters who is serving first and other match data as above. Serves then change according to the rules of tables tennis.

The user needs to enter the following data relating to each point, serve type by player or opponent, opening shot type by player or opponent, and who won the point. When there is a service fault, there is no opening shot. Using these variables, the system can produce all the needed statistics.

The serve types are as follows,

1. Pendulum
2. Backhand
3. Tomohawk
4. Punch
5. Reverse pendulum
6. Fault

There needs to be an ability to add new serves as required.

The opening shot types include

1. Forehand loop
2. Backhand loop
3. Forehand flick
4. Backhand flick
5. Forehand loop error
6. Backhand loop error
7. Forehand flick error
8. Backhand flick error
9. Push Error

Each opening shot can be played by the player or opponent and the user needs to indicate this through user data entry.

The user should be able to fix a data entry error, and rollback a point entry error.

The system should automatically select who is serving.

The match is best of 3, 5 or 7 games. The default should be 5 games.

After each point is entered, the default values for data entry should be the same as the previous point to save having to enter the same data multiple times when recurrence happens.

The data should be stored in a SQLite database for local storage.

The system should be created in Python.

with:

1. Tkinter for GUI
2. SQLite for local database storage
3. Pandas for data manipulation
4. Matplotlib or Plotly for data visualization

The user interface is to be created using Tkinter. Detailed guidance will be needed to build.

The current game score, historical games score of previous sets in the match, player names and opponent names will need to be dynamically displayed in the GUI.

The last 5 points, opening shots played, who played them and the result should also be shown in the GUI as point history.

The game ends when the player wins the best of match. The match will be best of 3 games, 5 games or 7 games.

Confirmation of the end of game should be asked.

You will be required to write the code, build the database, and assist with GUI creation.

I will need step by step assistance as I am a novice coder.

You will suggest the statistics that may be useful to extract and visualisations.

Is there anything else you need to help me build this application.