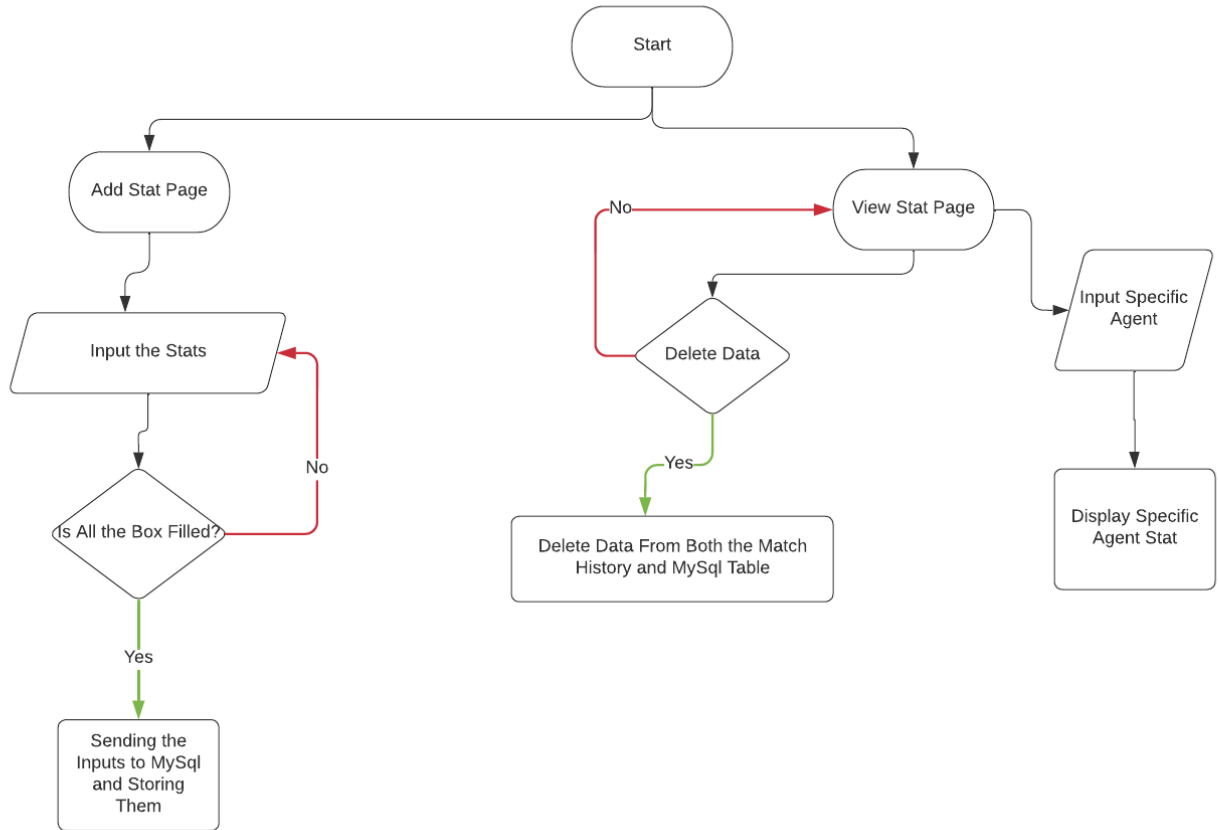


## Criterion B: Design

### Processing: Flowchart

- Shows the functionality from when the program is started.



### Development: Class Structure and MySql Table

- Shows what class the program will include and its attributes.
- Shows methods that will be created in the class.
- Shows what data will be stored in the MySql table.

UserStat Class
agent: String kill: int death: int assist: int win: boolean gameNumber: int id: int dfSharp: DecimalFormat
UserStat() UserStat(String, int, int, int, boolean) UserStat(String, int, int, int, boolean, int, int) getAgent(): String getKill(): int getDeath(): int getAssist(): int getWin(): boolean getID(): int getDataAsString(): String array calcKDA(): double

MySql Table		
NN	pKill	INT
NN	death	NN
NN	assist	INT
NN	win	TINYINT
NN	agent	VARCHAR(45)
PK	id	INT

### Development: Plan

1. Creating a database for the inputted stats
  - a. Creating a table that holds all the data inputted by the user.
  - b. Returning data within the database to the program.
2. Creating a UserStat class
  - a. Creating a class that will be used in order to store the data within the program and a method to calculate KDA.
  - b. Returning the stats to be used in the program.
3. Create UI using NetBean
  - a. Creating "Add Stat" and "View Stat" pages.
  - b. Adding a text box for the user to input data in.
  - c. Displaying stats and graphs according to the agent being selected.

Kill:

Death:

Assist:

Win:

Agent:

Graph

Match History

Total Kill:

Total Death:

Total Assist:

KDA

Total Win:

Total Game:

Win Rate:

Character:

**Test Plan:**

Action to Test	Method of Testing	Expected Result
The users will be able to see their match history.	Using tables to display match history and see if the data that are displayed is correct.	The table displays the data in a match history way, the newest data on top.
The users will be able to see their KDA graph.	Creating a panel and using it to display a KDA graph and	The panel displays the KDA graph correctly.

	see if it is displaying correctly.	
The users will be able to see summarized stats such as their kill death ratio, win rate etc.	Using labels in order to display each summarized stat and see if it is displaying correctly.	The labels display each of the summarized stats correctly.
The users will be able to choose stats according to the specific agent they want to see.	Creating a combo box so that the user can use it to select a specific agent they want to see.	The user can choose an agent from the combo box,
The graph and stats summary that is displayed corresponds to the specific agent the user selected.	Selecting an agent and seeing if the correct data is displayed.	The data being displayed is corresponding with the agent being selected.
The user will be able to delete/remove a match from match history.	Creating a button that the user can click to delete a match from match history and see if the deleted data is removed both in the match history table and MySQL table.	The deleted data is removed from both the match history table and MySQL table.
The user will be able to refresh the stat page in order to see new data	Creating a button that will refresh the page and see if the new data is displaying correctly.	The new data is displayed correctly after clicking the refresh button.
The user will be able to store data in MySQL	Add data from the program and check if the data is stored in MySQL table.	The data added from the program is correctly stored in the MySQL table.

Word Count: 446 (Does not include bullet points)