***DART GAME MANAGEMENT SYSTEM***

**ABSTRACT**

This is a system to help manage dart games played in the club.

The Dart Game Management System aims to address the current issue in manual scorekeeping and game management of the game. Darts involves throwing pointed projectiles at a circular target with the aim of hitting specific areas to score points.. The game has a long history and is enjoyed by many people. However, the manual process of tracking scores and managing games can be time-consuming, prone to errors, and can detract from the overall gaming experience.

In manual dart scoring, each player takes turns throwing three darts at the board. The points are awarded based on where the darts land on the board. The outermost ring of the board scores 1 point per dart, the next ring in scores 2 points per dart, and so on until the center of the board, known as the bullseye, which scores the highest points. The points are totaled after each turn, and the first player to reach the predetermined score wins the game.

The Dart Game Management System will improve the gaming experience by computerizing the scorekeeping and game management processes. With the system, players can register, create profiles, and participate in dart games with ease. The system will provide real-time feedback to players during games, such as score updates and performance metrics, making the game more interactive and engaging. The system will also generate leaderboards and statistics to track player performance and progress over time, adding an element of competition and motivation to the game

The system will provide the club manager with better visibility and control over the dart games played making the gaming operations more efficient and even satisfying the customers/gamers more .With this, the manager can make more informed decisions that enhance the gaming experience of the players and improve the club’s profitability..

In conclusion, the Dart Game Management System is a valuable tool that will revolutionize the game of darts. By automating the scorekeeping and game management processes, the system will improve the accuracy, speed, and overall gaming experience for players maximizing on the end goal which is revenue.

**OBJECTIVES**

* To automate the management of the dart game facility .This will help reduce the need for manual book-keeping.
* To improve the user experience of the players and managing staff by providing real time information and statistics together with the payment options.
* Enhance the accuracy of the player statistics and scores.
* To provide staff with an easy-to-use interface for managing game schedules, scores, and payments.
* Improve communication among the players staff and management.
* Increase the revenue through better management and utilization of facilities.
* Enabling future expansion to support additional features.

**ACTUAL OBJECTIVES**

***PROJECT ENTITIES***

* Players
* Dart Boards
* Games
* Payments
* Staff/Manager

***PLAYERS***

* Create a new player record in the database.
* Retrieve the details of a player in the database.
* Retrieve a list of all players in the database.
* Update the details of an existing player record.
* Delete a player record in the database.
* Retrieve the top scoring players
* Retrieve number of wins for a specific player.

***DART BOARD***

* Create a new boards record in the database.
* Retrieve a list of all dart boards in the database.
* Retrieve the details of a specific dart board,
* Update the details of a specific dart board in the database.
* Delete a dart board record in the database.

***GAMES***

* Create a new game record in the database.
* Retrieve a list of all games in the database.
* Retrieve the details of a specific game in the database.
* Update the details of a specific game in the database.
* Delete a game record in the database.
* Retrieve the number of games played at a specific dart board.

***PAYMENTS***

* Create a new payments record in the database.
* Retrieve all the payments in the database.
* Retrieve the details of a specific payment record in the database.
* Update the details of an existing payment record.
* Delete a payment record from the database.

***STAFF/MANAGER***

* Create a new staff record in the database.
* Update the details of an existing staff record.
* Delete a staff record from the database.

**SPECIFICATIONS**

* The system should be able to track scores and statistics for individual players and teams.
* The system should include a feature that allows the players to view their statistics and the trends in performance.
* The system should have a database system to store the necessary data.
* The system should be able to handle a large number of users and data over time.
* The system should be intergratable with the POS for inventory management via APIs.
* The system should be responsive to cater for all screen sizes.
* The system should have a backup system to ensure data is not lost in case of a system failure.
* The system should have a dashboard that displays the key performance indicators .This will be helpful to the administrators in monitoring the performance of the system.
* The system should include a leaderboard feature that displays the top players performance-wise based on games or tournaments played.

**ACTUAL IMPLEMENTATION**

**TECHNOLOGY STACK**

SYSTEM DESIGN – Figma

FRONT END – HTML, CSS, Vanilla/React JS

BACKEND – PHP/Laravel ,Node.js

DATABASE – SQL on MySQL

DATABASE DESIGN -

API DEVELOPMENT – RESTful, Express.js framework

VERSION CONTROL – Git on Github platform.

TESTING FRAMEWORKS – Jest

PAYMENT PROCESSING API – M-Pesa

DATA VISUALIZATION LIBRARY – Chart.js , D3.js

**DEVELOPMENT PROCESS**

**Version Control System**

Git will be used to manage code and collaborate with team members.

**Project Management**

Trello will be used to track progress and manage tasks.

**Communication**

The team members will communicate through Slack or Zoom.

Using Slack, different channels can be used for different topics.

**CODE DOCUMENTATION**

**Comments**

Comments will be used within the code to explain different functions and compone**nts.**

**Components Functionality**

An overview of the system architecture outlining the structure and the different functionalities of the system.

**User Manual**

A user manual should be created to provide instructions for using the different features of the system.

It should include screenshots and step-by-step instructions to help the users and the manager navigate the system very comfortably.

**PERFORMANCE TESTING**

**Load Testing**

A load test should be carried out to make sure that the system can handle a high volume of traffic without lagging or having any performance issues.

**Simulated User Load Testing**

During this testing the system will be subjected to different simulated user loads and various performance metrics recorded .These are ; response times and server usage.

**SECURITY TESTING**

**Vulnerability Scans**

Vulnerability scans should be carried out to identify potential security risks.

**RESULTS AND EVALUATION**

**HOW THE GAME IS PLAYED**

The game is played by two players.

The game’s objective is to score 501 points.

Each player starts the game with 501 points, and the goal is to get to exactly 0.

Each player has 3 tries per turn.

To win, the player must end by hitting a double.Its total when deducted from the current points should reach exactly 0.

If the last turn ends at 1 or below 0, this is called a bust and the player’s score is returned to where it was before the turn.

If the score ends at 0 after a double then the player has won the game.

**HOW THE SYSTEM WILL MANAGE THE GAME**

The system should have a menu and in the menu there is the kick-off button.

After clicking the kick-off button, the system should prompt the players to enter their names and any other important information.

The information is saved and the system then prompts the players to start through another button.

After hitting the start button, the game starts and each player is given 501 points.

The system flags off player 1 to start and he/she has 3 tries per turn.

After the third try, the total points for the three tries are added and the total subtracted from the grand total which is currently 501.

The new score is now displayed under player 1’s name and that becomes the current score.

The system then flags off player 2’s turn.

The process is similar to player 1’s.

The system should take these turns until one player meets all the necessary rules to win a game.

The system should also check for busts.

A ‘bust’ occurs when one of the following happens ;

-A player reaches exactly 1 or below 0.

-A player reaches 0 but not by a double.

In case of a bust, the player’s score is restored to where it was before the turn and the next player is signaled to player his/her turn.