Group assignment

|  |  |
| --- | --- |
| **Authors:** | Juha Duong |
|  | Eerik Vainio  Chen Jiao  Elina Maunula | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **Date:** | 03.04.2024 | Version 1.0 |
|  |  |  |

1. Description

Read the rules of Darts (search the Net). Concentrate only on the games “301 up” and “501 up”. A game is also called a leg. Legs are played in sets. A set is won by the player who wins more legs the other. For example, 3 leg wins is enough to win a set of 5 legs.

Create a single-page web application that can be used keep score in Darts. The application should provide at least the following functionality. Before the set starts: To add players (2), select the game type (301 up or 501 up), and select the set size (how many legs are played at most to find a winner). For each leg: Keep the score of each turn of each player and maintain each player’s total score. At the set level: Keep record of the number won legs of each player. Please also note at the application should automatize some operations, like resetting the leg scores after every leg, and declaring the winner.

Nothing needs to be saved permanently. It is alright if the application loses its state in the browser refresh.

The evaluation criteria used is the following:

- The application fulfils the given requirements (4 points)

- The application makes it possible to view and list some kind of objects from the application state – like the turns in the Darts score keeper application (2 points)

- The application makes it possible to add, modify, and remove those objects to and from the application state - like adding, modifying, and removing turns in the Darts score keeper application (3 points)

- The application makes it possible to view different kinds of real-time aggregations (related to the addings and other modifications) (sums, averages, …) that are calculated over the objects is the application state - like players’ legs’ total scores or count of players’ won legs in the Darts score keeper application. (3 points)

- The application is divided into components in a justified way (2 points)

- The components are functional components (1 point)

- React hooks are used (2 points)

- Context API is used to manage application state (2 points)

- The application functions without exceptions (2 points)

- The application is styled with StyleSheets and looks good and specific (3 points)

- Git is used by the group (1 point)

- create\_react\_app is used (1 point)

- Other suitable development tools are used. (1 point)

- The additional React or JavaScript features not covered by the instructor are used. (3 points)

2. Suggested approach to assignment

Create a single-page web application that can be used keep score in Darts. The application should provide at least the following functionality. Before the set starts: To add players (2), select the game type (301 up or 501 up), and select the set size (how many legs are played at most to find a winner).

For each leg: Keep the score of each turn of each player and maintain each player’s total score. At the set level: Keep record of the number won legs of each player. Please also note at the application should automatize some operations, like resetting the leg scores after every leg, and declaring the winner.

* Divide the functionalities to own modules (players, game type and set leg size, keeping the score each turn and maintaining total score).
* Player module: Juha
* Game type and set leg size module:
* Keeping the score and maintaining total score:
* Something else?
* Use separate branch to work on SPECIFIC module (no overlapping with other modules)
* If not sure where to begin, check the expense-tracker exercise