# Layout

|  |  |
| --- | --- |
| CRITERIA | MEETS SPECIFICATIONS |
| Game Chosen | The chosen game has either multiple amounts of points that can be scored, as in American football, or multiple important metrics to track, such as fouls, outs, and innings in baseball. |
| Overall layout | App is divided into two columns, one for each team. |
| Column contents | Each column contains a large TextView to keep track of the current score for that team.  Optionally, a second TextView to track another important metric such as fouls can be added. |
| Score buttons | Each column contains multiple buttons. The buttons must track either:  Each track a different kind of scoring  Or  Each track a different metric (one score, the other fouls, for instance). |
| Reset button | The layout contains a ‘reset’ button. |
| Best practices | "The code adheres to all of the following best practices:  Text sizes are defined in sp  Lengths are defined in dp  Padding and margin is used appropriately, such that the views are not crammed up against each other." |

# Functionality

|  |  |  |
| --- | --- | --- |
| CRITERIA | MEETS SPECIFICATIONS | CRITERIA |
| Errors | The code runs without errors. | Errors |
| Score Button Function | Each score button updates the score TextView in its column by adding the correct number of points. | Score Button Function |
| Reset Button Function | The reset button resets the scores on both of the score TextViews. | Reset Button Function |

# Code Readability

|  |  |
| --- | --- |
| CRITERIA | MEETS SPECIFICATIONS |
| Naming conventions | "Any classes are named after the object they represent. All variables are named by their intended contents.  All methods are named by their intended effect or in the style required by a call back interface." |
| Code style | "There are no unnecessary blank lines. One variable is declared per declaration line.  The code within a method is indented with respect to the method declaration line." |