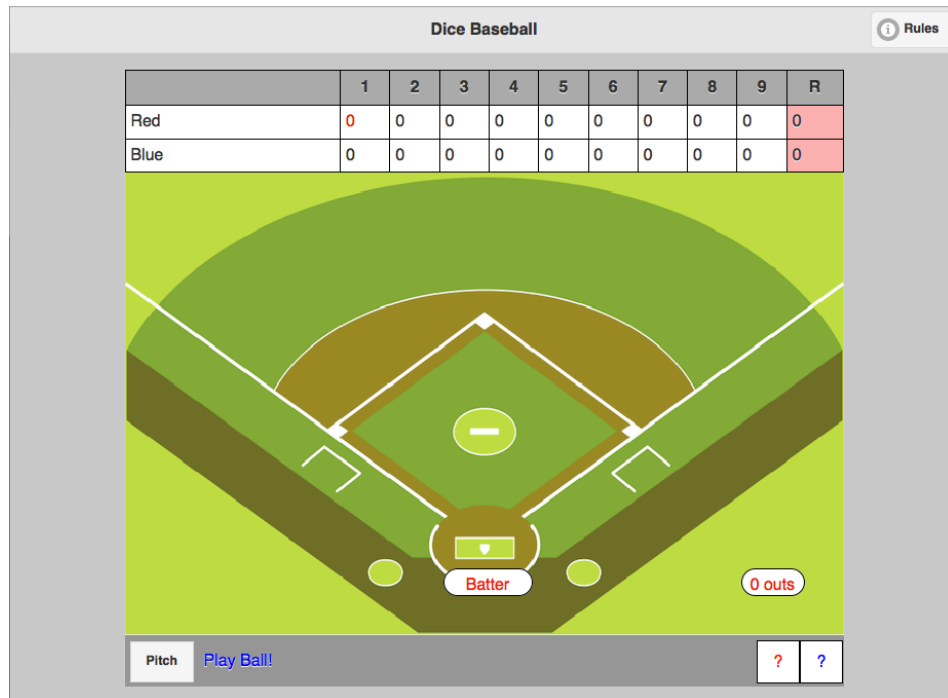
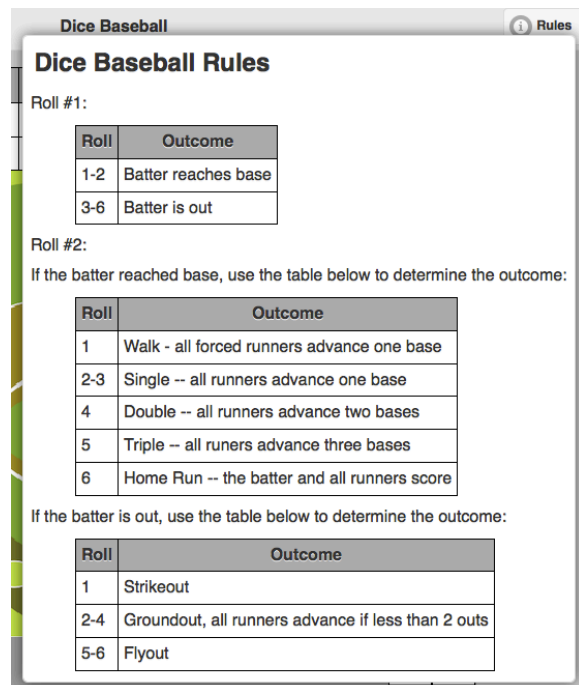


For this assignment, I would like you to create a simple dice baseball game. When your page is first loaded, your game screen should look like image shown below. Each time the user clicks on the “Pitch” button, two 6-sided die will be rolled. A dice roll should be represented using animation, for example each die should cycle randomly through a few values before they settle on a final value.



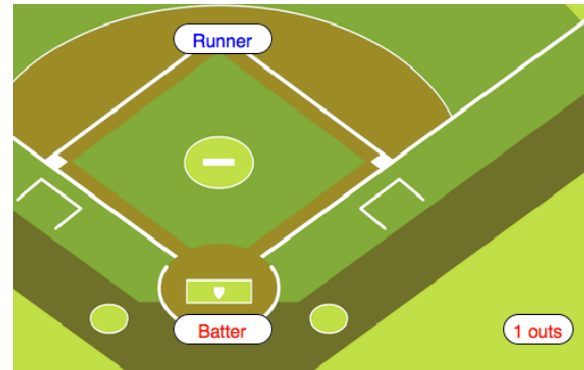
The outcome of the pitch is displayed next to the “Pitch” button and is determined by the game rules. The rules for dice baseball are pretty simple. The batting team rolls two 6-sided dice and the outcome is determined using the tables shown on the right. These tables should be displayed in your app (as a popup) when the user clicks on the “Rules” toolbar button located in the top-right of the header.

Based on these rules, if the first roll is a 1 or a 2, the batter reaches base and the second roll determines how they reach base (e.g., if the second role is a 5 the batter hits a triple). However, if the first role is greater than 2, the batter is out and the second roll determines how they got out (e.g., if the second role is a 1 they strikeout).



Notice that the rules also specify how runners should advance when the batter reaches base or if there is a groundout. For example, if there is a runner on first and second, and the batter hits a double, all runners advance two bases. This means that the runner on second will score, and the runner on first ends up on third. On the other hand, if there is a runner on first and third, and the batter walks, only forced runners advance. This means that the runner on first will go to second, but the runner on third remains on third.

Each time a runner reaches base, they should appear on the field. In addition, the number of outs in the current inning should be displayed. The image on the right shows what the field should look like if there is 1 out and a runner on second base.



Your app should also keep track of what team is currently up to bat, what the current score is, and how many runs scored each half-inning. All of this information should be displayed along the top of the game screen. For example, in the image shown below, the Red team is currently up to bat in the top of the second, the Blue team is winning 3-0, and the Blue team scored 3 runs in the bottom of the first.

	1	2	3	4	5	6	7	8	9	R
Red	0	0	0	0	0	0	0	0	0	0
Blue	3	0	0	0	0	0	0	0	0	3

There are two situations that can result in the game ending: (1) The away team makes three outs in the top of the 9<sup>th</sup> and the home team is leading or (2) the home team makes three outs in the bottom of the 9<sup>th</sup>. Unlike real baseball, the game can end in a tie. When the game ends, a message should be displayed (as shown in the image to the right), and any clicks on the “Pitch” button should be ignored. Refreshing the page should restart the game.



I will provide you with the image that you should use to represent the baseball field. Your app should use HTML, JavaScript, JQuery, JQuery Mobile, and the basic MVC architecture we learned about in class. The game will be demonstrated in class to make sure everyone understands how it works.

You should consult the rubric for details about how this assignment will be graded. Please submit your assignment to your private GitHub repository for this class.

Enjoy!