

Task 1 Objective:

To create an online news app that makes asynchronous calls to API – BBC News API and displays the news results.

Code:

a. Html page

The launch page called home.html. This page toggles between Login and Register. The login behind login and register is not implemented as it is not in the scope.

```
index.html x tictactoe.js x home.html x
10
11 </head>
12 <body>
13 <div class="container" id="home" >
14 <div class="col-md-6 col-md-offset-3" id="register" style="...">
15 <form action="news.html" style="...">
16 <h1>Register</h1>
17 <p>Please create an account to access online news</p>
18 <hr>
19 <b>Email</b>
20 <input type="text" placeholder="Enter Email" name="email" required>
21 <b>Password</b>
22 <input type="password" placeholder="Enter Password" name="pass" required>
23 <b>Repeat Password</b>
24 <input type="password" placeholder="Repeat Password" name="pass-repeat" required>
25 <button type="submit" class="signupbtn">Register</button>
26 <p>Already have an account! <a onclick="toggle()" style="...">Login</a></p>
27 </form>
28 </div>
29
30 <div class="col-md-6 col-md-offset-3" id="login">
31 <form action="news.html" style="...">
32 <h1>Login</h1>
33 <p>Please login to access online news</p>
34 <hr>
35 <b>Email</b>
36 <input type="text" placeholder="Enter Email" name="email" required>
37 <b>Password</b>
38 <input type="password" placeholder="Enter Password" name="pass" required>
39 <button type="submit" class="loginbtn">Login</button>
40 <p>Don't have an account! <a onclick="toggle()" style="...">Register</a></p>
41 </form>
42 </div>
43 </div>
```

The search page(news.html) consists of two criteria by which a user can search news. They are search topic and news source. At least one of the search criteria needs to be selected. The news is displayed as heading, time and image. On clicking the image, the news article opens in a new tab.

```

index.html x  tictactoe.js x  home.html x  news.html x
15 </head>
16 <body onload="findResources()">
17   <div class="container">
18     <div class="row">
19       <div class="col-md-10 col-md-offset-1">
20         <div class="panel panel-default">
21           <div class="panel-heading" style="...">
22             <div class="row">
23               <div class="col-md-1">
24                 
26               </div>
27               <div class="col-md-3">
28                 <h3 class="text-center">News Search</h3>
29               </div>
30             </div>
31           </div>
32           <div class="panel-body">
33             <br>
34             <span><b>Search By Topic:</b></span>
35             <input id="topicTextbox" type="text" class="form-control" placeholder="Search by topic.." />
36             <br>
37             <span><b>Search By Source:</b></span>
38             <select class="form-control" id="sourceDropdown"></select>
39             <br>
40             <div class="col-md-4 col-sm-offset-4">
41               <button class="btn btn-md btn-info" onclick="searchNews()">Search</button>
42             </div>
43             <div class="col-lg-12 col-md-12 col-sm-12 text-center" id="output"></div>
44           </div>
45         </div>
46       </div>
47     </div>
48   </div>

```

b. news.js (Javascript)

This javascript contains all the code behind toggle of login/register page, API call using ajax, error messages etc.

```

index.html x  tictactoe.js x  home.html x  news.html x  news.js x
1 //toggle between login and register
2 function toggle() {
3   var loginBlock = document.getElementById("login");
4   var registerBlock = document.getElementById("register");
5
6   if (loginBlock.style.display === "none") {
7     loginBlock.style.display = "block";
8     registerBlock.style.display = "none";
9   } else {
10    loginBlock.style.display = "none";
11    registerBlock.style.display = "block";
12  }
13 }
14
15 var news_url = "https://newsapi.org/v2/sources?";
16 var top_headlines_url = "https://newsapi.org/v2/top-headlines?";
17 var developer_key = "50f74aaace1c4c81a3b32c63ef8cf5d7";
18
19 function findResources() {
20   $.ajax({
21     url: news_url+"apiKey="+developer_key
22   }).done(function(response) {
23     var sources = response.sources;
24     if (sources.length !== 0) {
25       var sourceDD = document.getElementById("sourceDropdown");
26       sourceDD.options[sourceDD.options.length] = new Option('Select Source', '');
27       for (var i = 0; i < sources.length; i++) {
28         sourceDD.options[sourceDD.options.length] = new Option(sources[i].name, sources[i].id);
29       }
30     } else {
31       alert("Unable to load source list!!!");
32     }
33   });
34 }

```

```

36 function searchNews(){
37     var topic = document.getElementById("topicTextbox").value;
38     var selectedSource = document.getElementById("sourceDropdown").value;
39     // check if user has entered both topic and source or either one.
40     if (topic && selectedSource) {
41         var parameters = "q=" + topic + "&sources=" + selectedSource;
42     } else if (topic) {
43         var parameters = "q=" + topic;
44     } else if (selectedSource) {
45         var parameters = "sources=" + selectedSource;
46     } else {
47         displayError();
48         return;
49     }
50     // create request url
51     var url = top_headlines_url + parameters + "&apiKey=" + developer_key;
52     $.ajax({
53         url: url
54     }).done( function(response){
55         var articles = response.articles;
56         if (articles.length !== 0) {
57             var output = "";
58             for (var i = 0; i < articles.length; i++) {
59                 output += "<h3>" + articles[i].title + "</h3>";
60                 output += "<p>" + (new Date(articles[i].publishedAt)).toUTCString() + "</p>";
61                 output += "<a href='" + articles[i].url + "' target='_blank'" + "<img src='"
62                     + articles[i].urlToImage + "' class='img-responsive img-thumbnail'" + "></a>";
63                 output += "<hr>";
64             }
65             displaySuccess(output);
66         }else{
67             displayNoResultError();
68         }
69     });

```

Output:

When the page loads.

The screenshot shows a web browser window with the address bar displaying a localhost URL. The page content is a login form titled "Login". Below the title, it says "Please login to access online news". There are two input fields: "Email" with the placeholder text "Enter Email" and "Password" with the placeholder text "Enter Password". Below these fields is a blue "Login" button. At the bottom of the form, there is a link that says "Don't have an account! Register".

Toggle when user clicks on Register

Register

Please create an account to access online news

Email

Password

Repeat Password

Already have an account! [Login](#)

Login page logic not implemented, still all the mandatory fields needs to be entered.

Login

Please login to access online news


Email

Password

Don't have an account! [Register](#)

When the user logs in or registers, the news page is loaded.

News Search




News Search

Search By Topic:

Search By Source:

At least one criteria needs to be selected.


 News Search

Search By Topic:

Search By Source:

Please select atleast one criteria.

A proper message is displayed when no results are found


 News Search

Search By Topic:

Search By Source:

No results found!

News being displayed on the page


 News Search

Search By Topic:

Search By Source:

Series win against South Africa doesn't happen every day - Shastri

Sat, 17 Feb 2018 01:38:45 GMT



When the user clicks on image, the article is displayed in new tab.

News Search x Series win against South x

www.cricbuzz.com/cricket-news/100449/series-win-against-south-africa-doesnt-happen-every-day-ravi-shastri

cricbuzz Live Scores Schedule Archives News Series Teams Videos Photos Rankings More

MATCHES RSA vs IND - IND Won AFG vs ZIM - AFG Won RSAW vs INDW - IND... TAS vs WA - Tea VIC vs QL - Live ALL

INDIA TOUR OF SOUTH AFRICA

Series win against South Africa doesn't happen every day - Shastri

Kaushik Rangarajan · Last updated on Sat, 17 Feb, 2018, 07:44 AM

Share Tweet Share



PLAYER SEARCH

Search Player GO

LATEST NEWS

-  Series win against South Africa doesn't happen every day - Shastri 2h ago
-  'Superman' Kohli doesn't want the headlines 2h ago
-  Aiden Markram preaches need for error correction 6h ago
-  Mandhana effusive in praise for senior pro Mithali Raj 7h ago

Task 2 Objective:

Tic-Tac-Toe game using Angular JS.

Code:

a. HTML page

The html page contains the board with angular js element ng-repeat. The page displays the current player when game is in progress. When the game is complete, it displays the results along with an option to play a new game.

```
index.html x
5      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.7/angular.min.js"></script>
6      <link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" rel="sty
7      <meta name="viewport" content="width=device-width, initial-scale=1">
8      <link href="css/stylesheets.css" rel="stylesheet">
9      <script src="js/tictactoe.js"></script>
10     </head>
11     <body ng-controller="TicTacToeCtrl">
12     <div class="container">
13     <h1>Tic Tac Toe</h1>
14     <div class="game">
15     <div class="board" ng-repeat="row in board">
16     <button type="button"
17     class="btn-lg"
18     ng-repeat="cell in row"
19     ng-disabled="!unplayed(cell)"
20     ng-class="cell.value"
21     ng-click="flip(cell)">{{ cell.value }}
22     </button>
23     </div>
24     <h4 id="turn" ng-show="!gameOver()">Player {{ currentPlayer }}'s Turn</h4>
25     <br>
26     <h2 id="result"></h2>
27     <button id=resetBtn
28     type="button"
29     ng-hide="!gameOver()"
30     class="btn-lg"
31     data-ng-click="reset()">New Game</button>
32     </div>
33     </div>
34     </body>
35     </html>
```

b. Javascript page

This page consists of all the game logic to set and reset the board when started. It checks if the game is over by checking rows, columns, diagonal and tie-case. It also contains the logic to flip a cell.

```
angular.module("TicTacToe", []).controller("TicTacToeCtrl", function($scope) {
    var unplayedCell = "?";
    var gameOverValue = "N";

    $scope.reset = function() {
        $scope.board = [
            [{value : unplayedCell}, {value : unplayedCell}, {value : unplayedCell}],
            [{value : unplayedCell}, {value : unplayedCell}, {value : unplayedCell}],
            [{value : unplayedCell}, {value : unplayedCell}, {value : unplayedCell}]
        ];

        $scope.currentPlayer = "X";
        $scope.winner = null;
        document.getElementById("result").innerHTML = "";
        gameOverValue = "N";
    };

    $scope.reset();
});
```

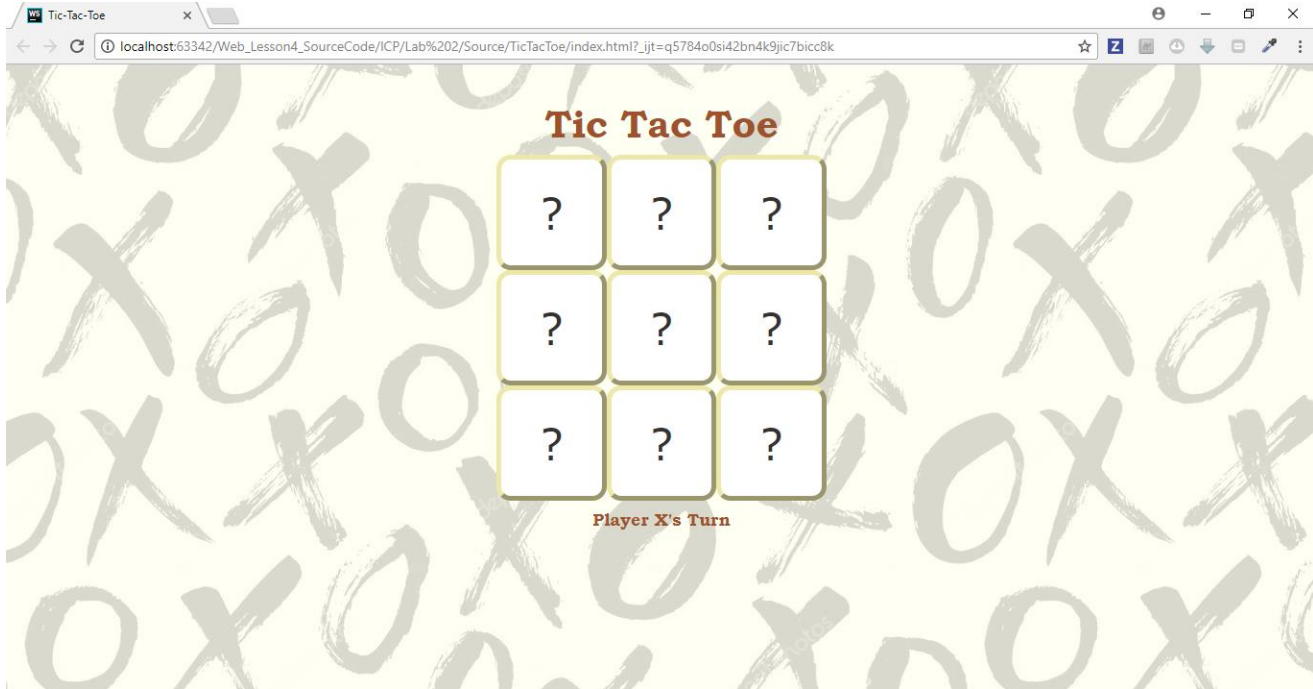
```
index.html x tictactoe.js x
20 $scope.checkGameOver = function() {
21     var cell = "";
22     //check rows
23     for(var i = 0; i < 3; i++) {
24         cell = $scope.board[i][0].value;
25         if(cell !== unplayedCell &&
26             cell === $scope.board[i][1].value &&
27             cell === $scope.board[i][2].value) {
28             $scope.winner = cell;
29             return true;
30         }
31     }
32     //check columns
33     for(var j = 0; j < 3; j++) {
34         cell = $scope.board[0][j].value;
35         if(cell !== unplayedCell &&
36             cell === $scope.board[1][j].value &&
37             cell === $scope.board[2][j].value) {
38             $scope.winner = cell;
39             return true;
40         }
41     }
42     //check diagonals
43     var k = 1;
44     cell = $scope.board[k][k].value;
45     if(cell !== unplayedCell &&
46         cell === $scope.board[k-1][k-1].value &&
47         cell === $scope.board[k+1][k+1].value) {
48         $scope.winner = cell;
49         return true;
50     }
51     else if(cell !== unplayedCell &&
52         cell === $scope.board[k-1][k+1].value &&
53         cell === $scope.board[k+1][k-1].value) {
54
55     }
56
57     $scope.unplayed = function(cell) {
58         return cell.value === unplayedCell;
59     };
60
61     $scope.gameOver = function() {
62         return gameOverValue === "Y";
63     };
64
65     $scope.flip = function(cell) {
66         if (!$scope.checkGameOver()) {
67             if ($scope.unplayed) {
68                 cell.value = $scope.currentPlayer;
69
70                 if ($scope.checkGameOver()) {
71                     if ($scope.winner === "X") {
72                         document.getElementById("result").innerHTML = "Player X won!";
73                     } else if ($scope.winner === "O") {
74                         document.getElementById("result").innerHTML = "Player O won!";
75                     } else {
76                         document.getElementById("result").innerHTML = "It's a Tie!";
77                     }
78                     gameOverValue = "Y";
79                 }
80             } else {
81                 if ($scope.currentPlayer === "X") {
82                     $scope.currentPlayer = "O";
83                 } else {
84                     $scope.currentPlayer = "X";
85                 }
86             }
87         }
88     }
89 }
```

c. css page

The css page contains code for styling and displaying the elements.

Output:

Home page. The game starts with player X's turn



After a tile is clicked, message displays "Player O's turn"



Diagonal win – Message displays who won. Also a new game button is enabled.



Horizontal win



Tie situation.

