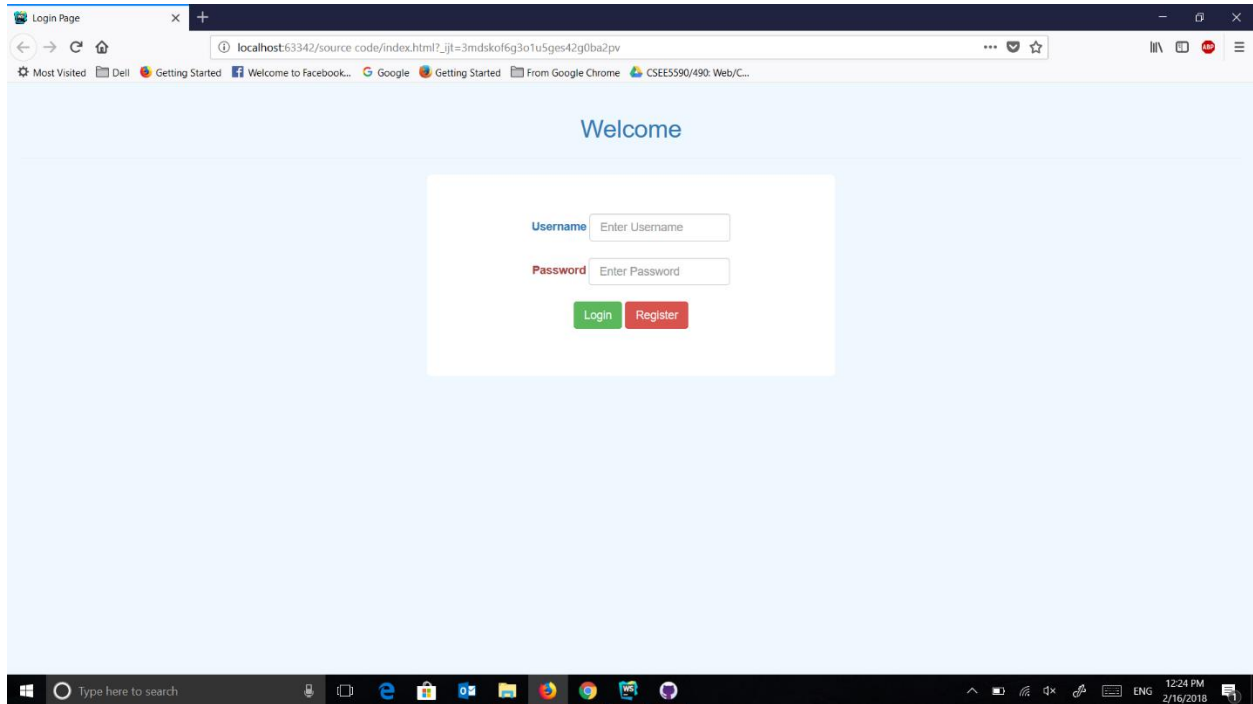


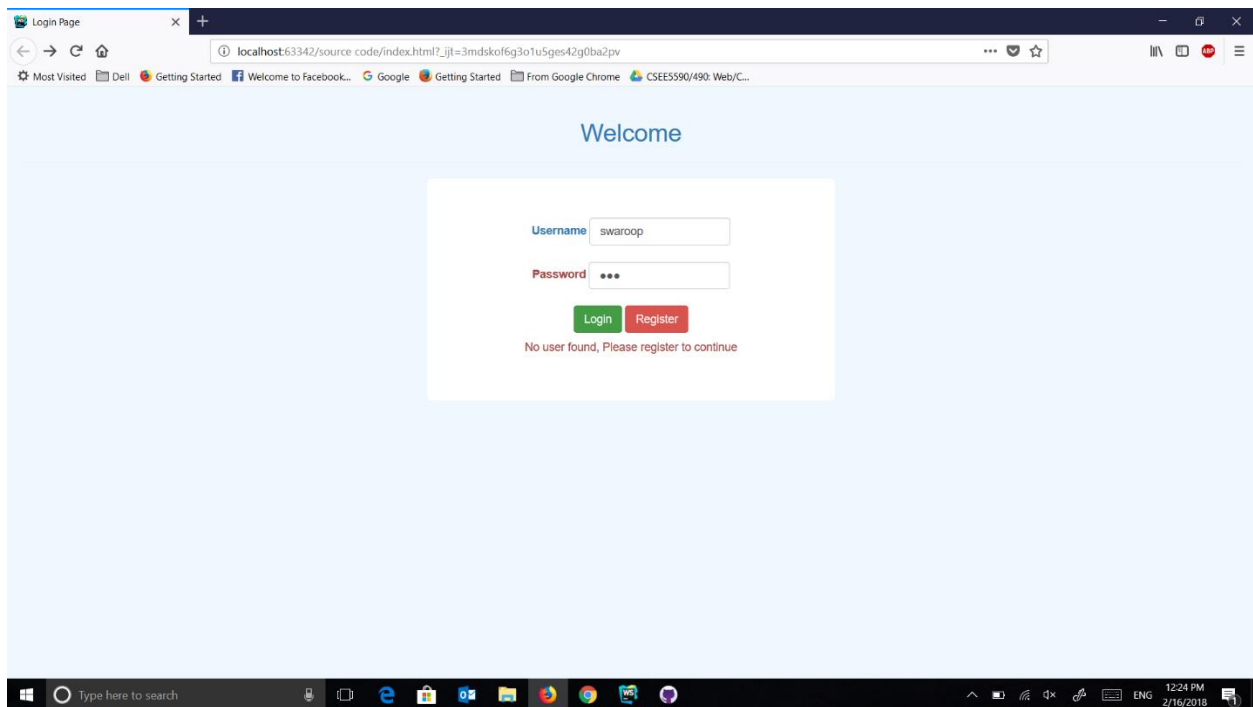
Lab Assignment #2

Part 1: News App

Created a login page with username and password.

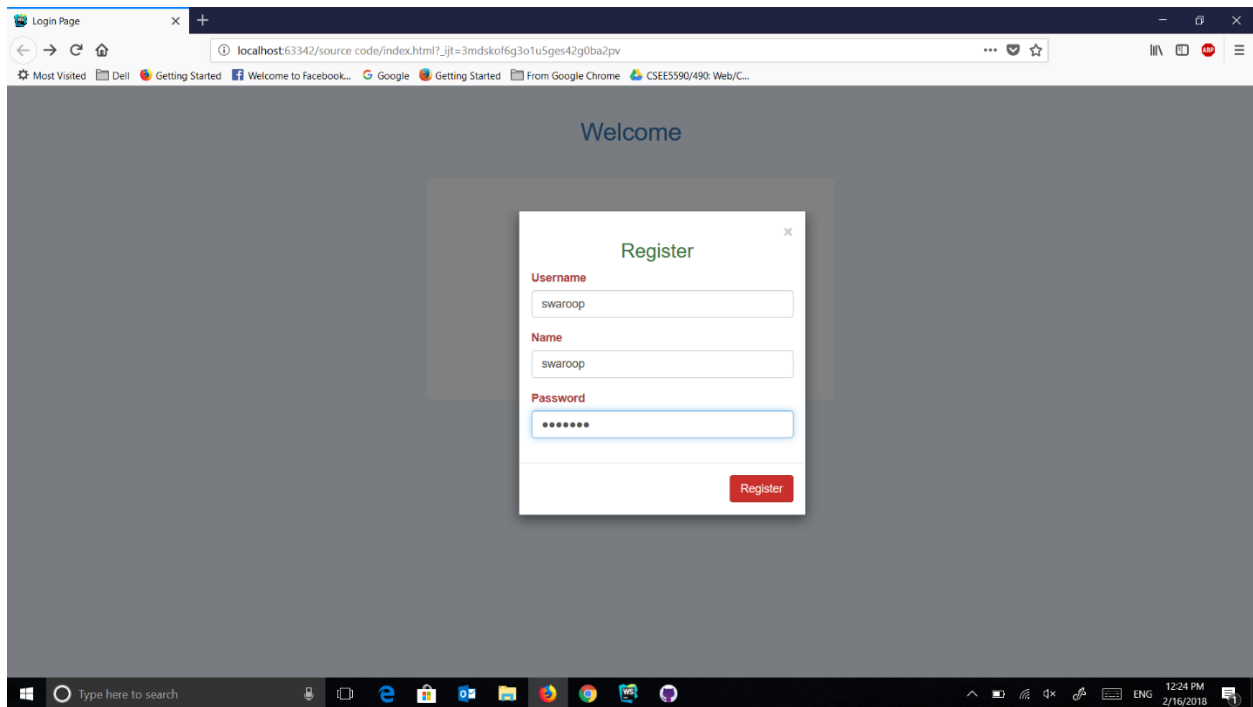


If username and password doesn't match, it will ask for registration

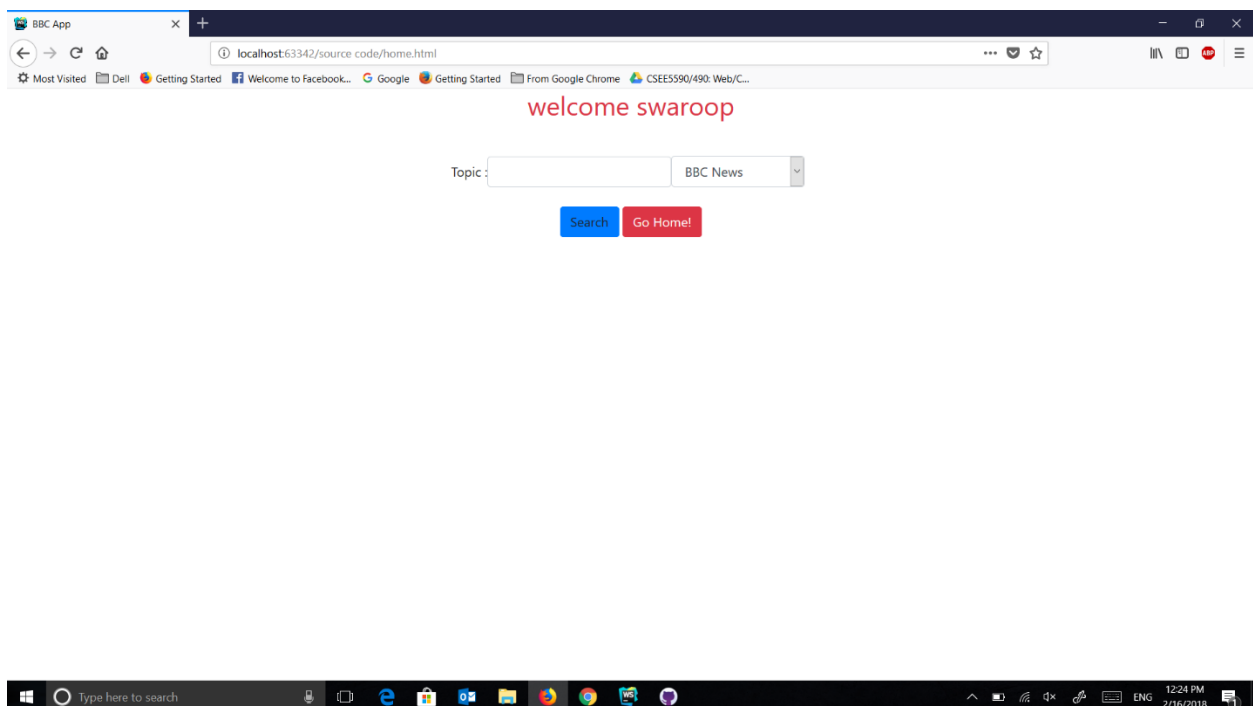


Lab Assignment #2

Registration Page

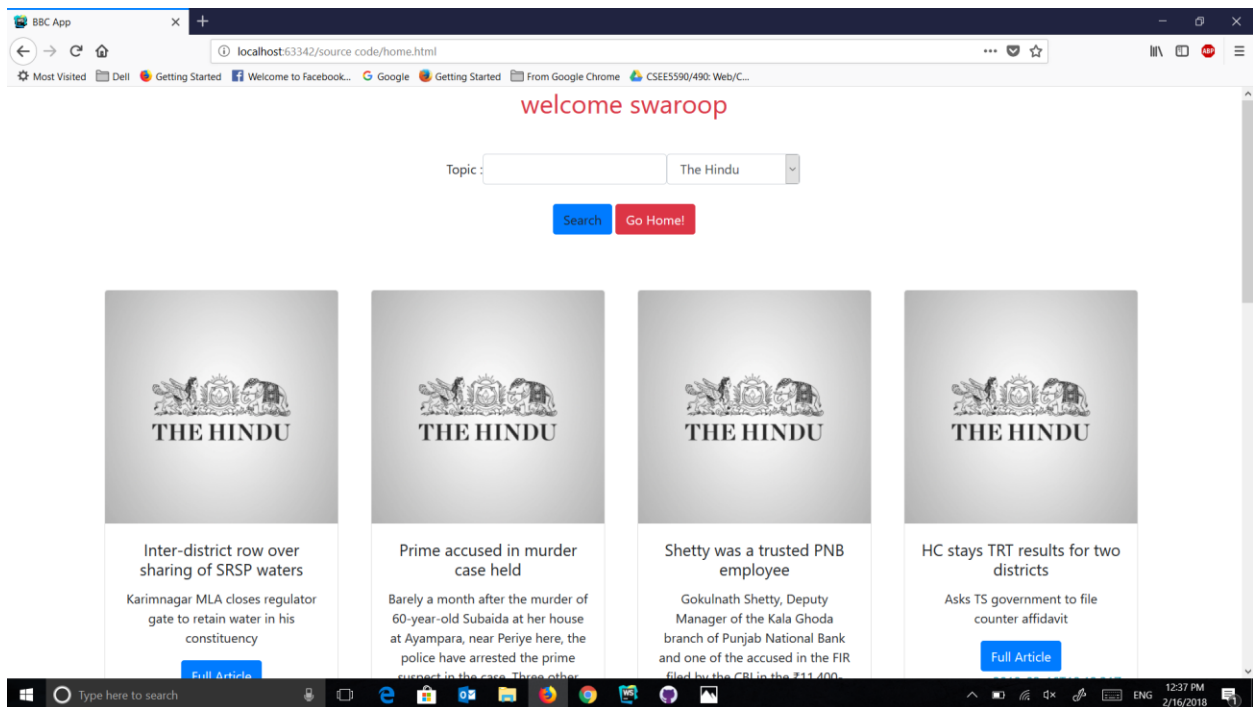
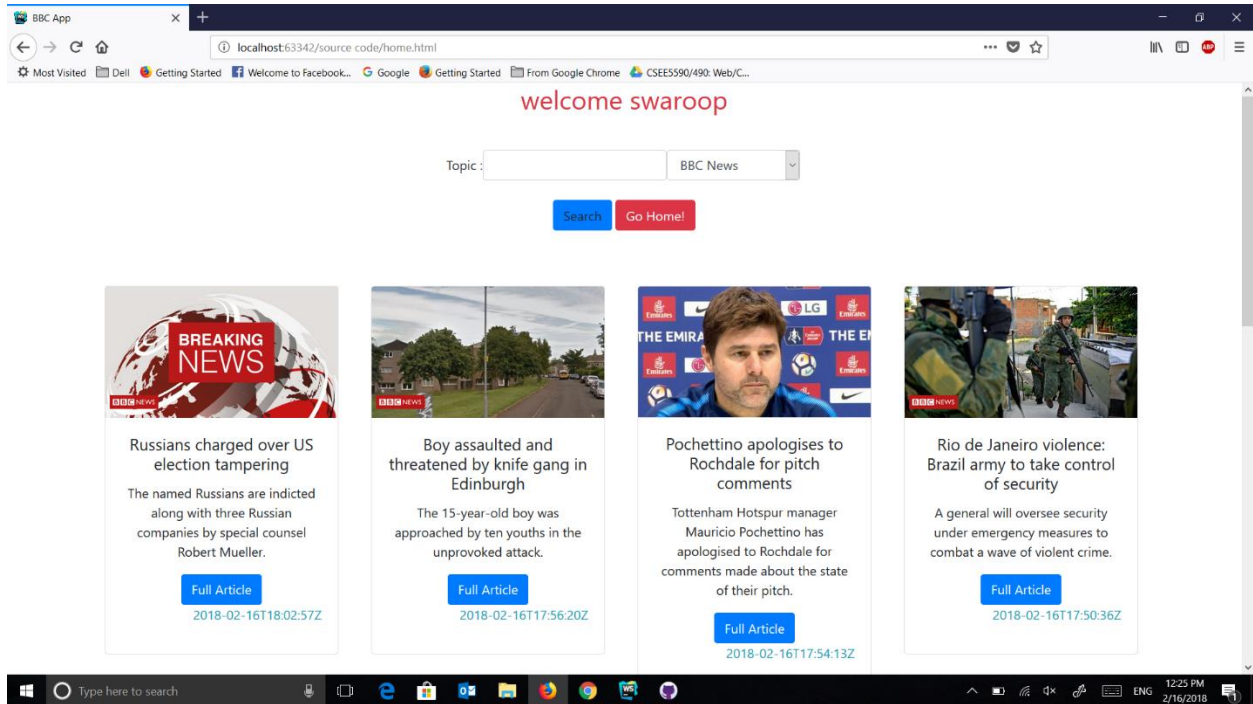


After Registration and signing in, it takes to the home page. which contains search bar to search a topic and dropdown for different news sources.

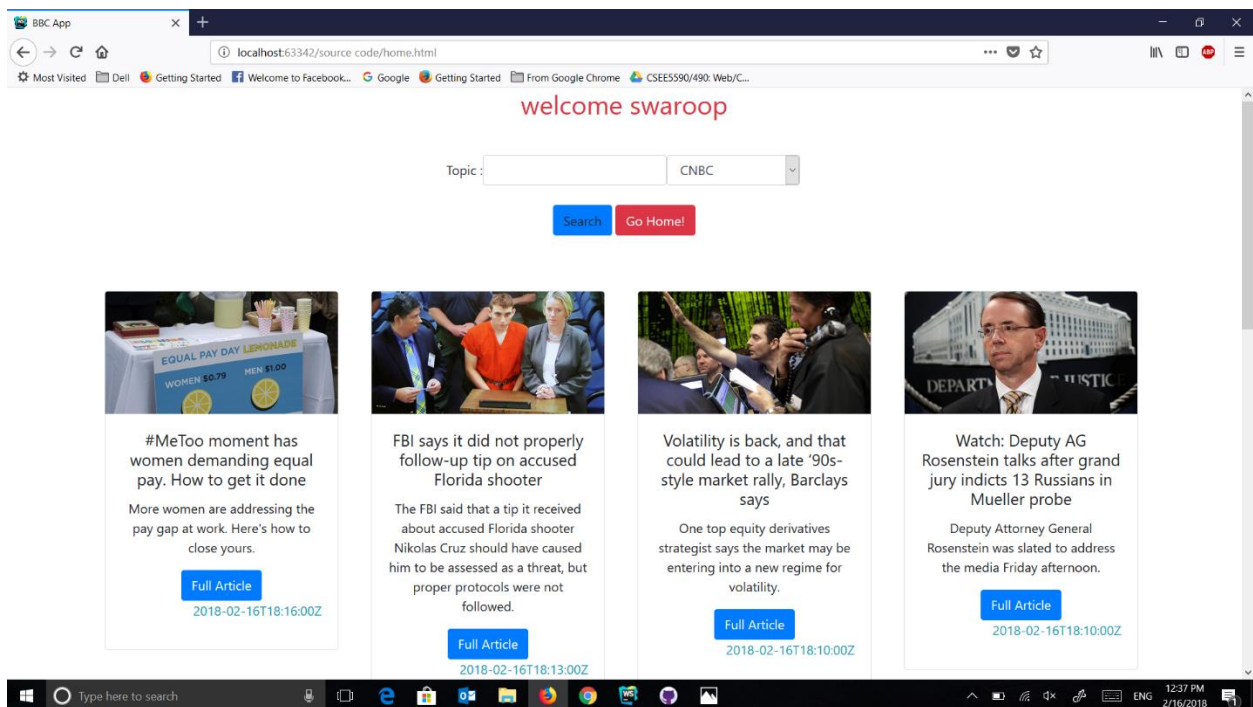
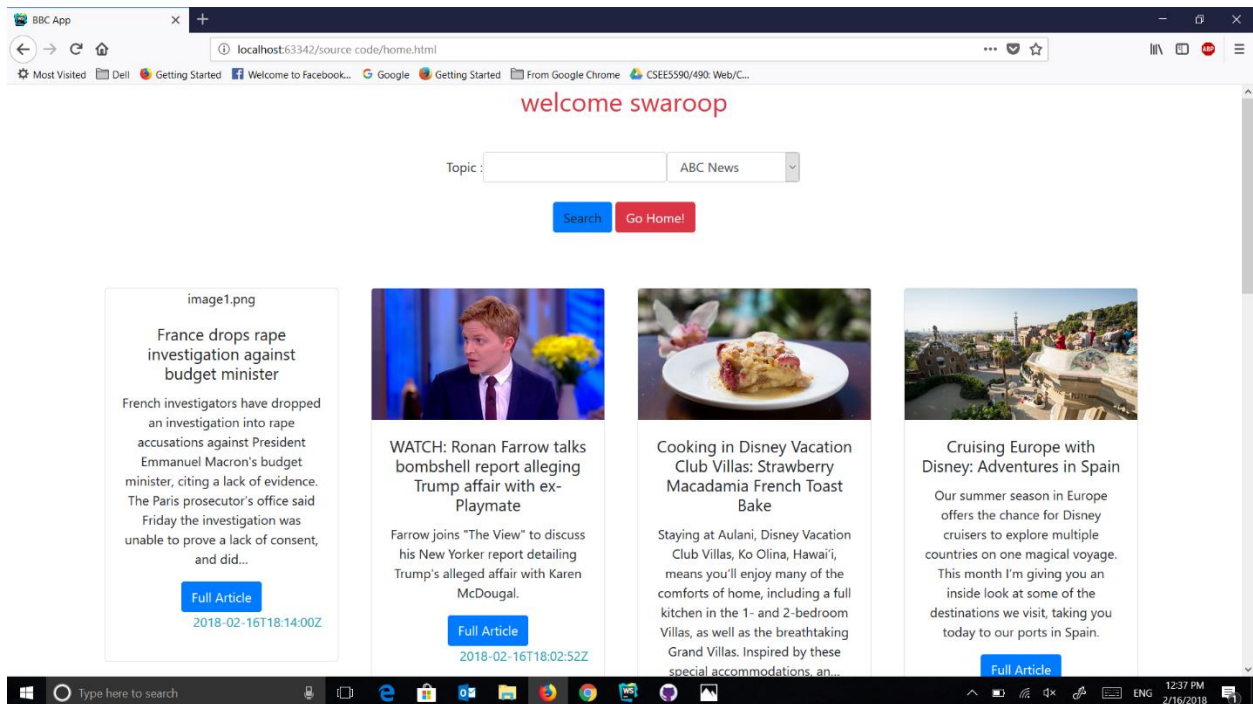


Lab Assignment #2

If we search without topic but with source, it will show the top news from the source



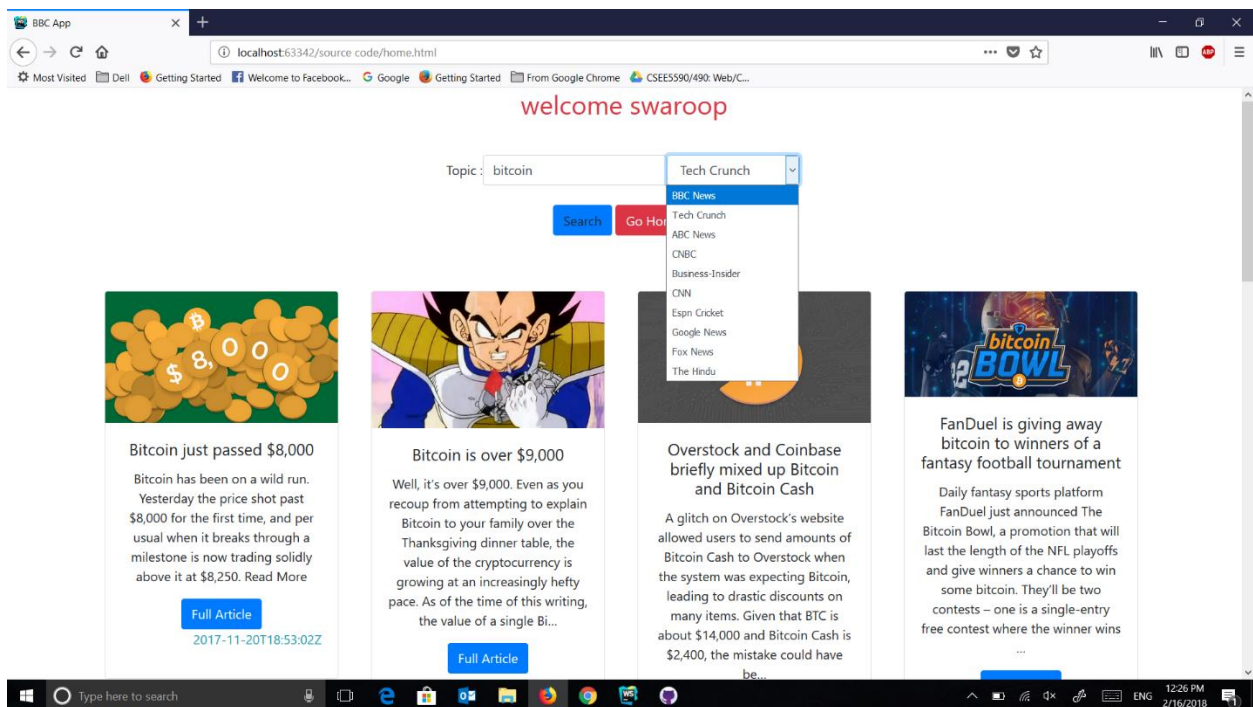
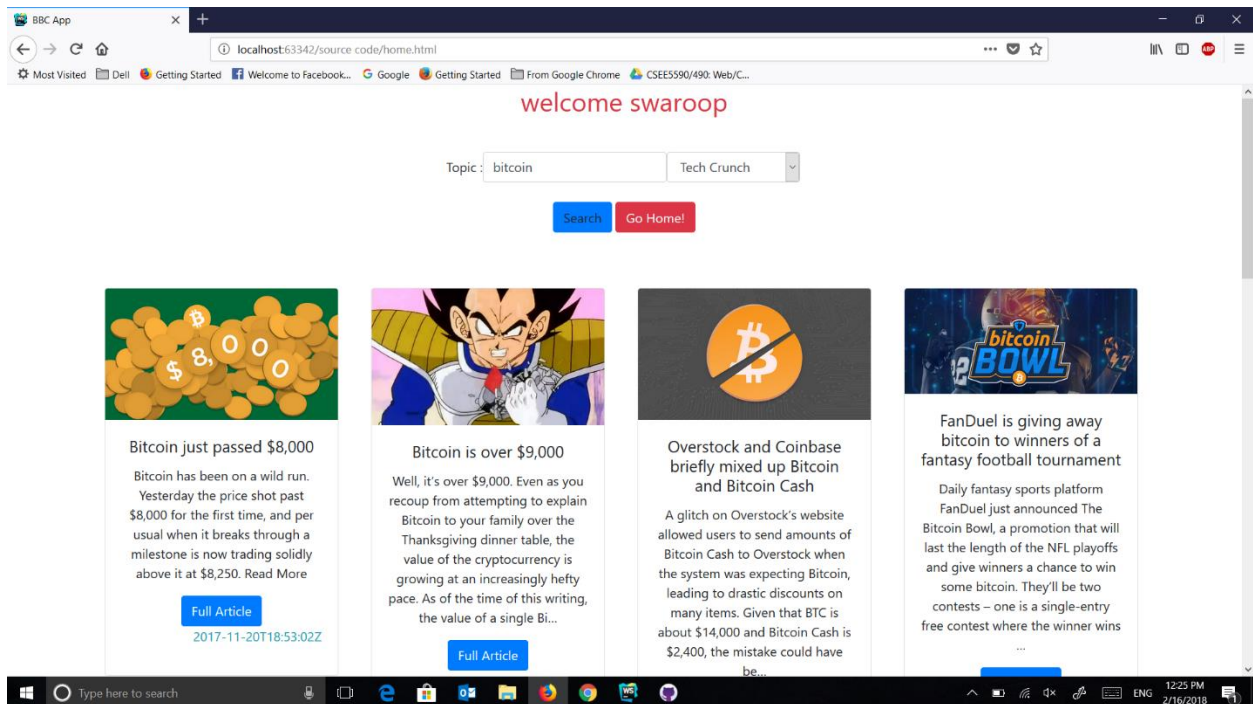
Lab Assignment #2



If you click on full article it open new tab with the full content of the article.

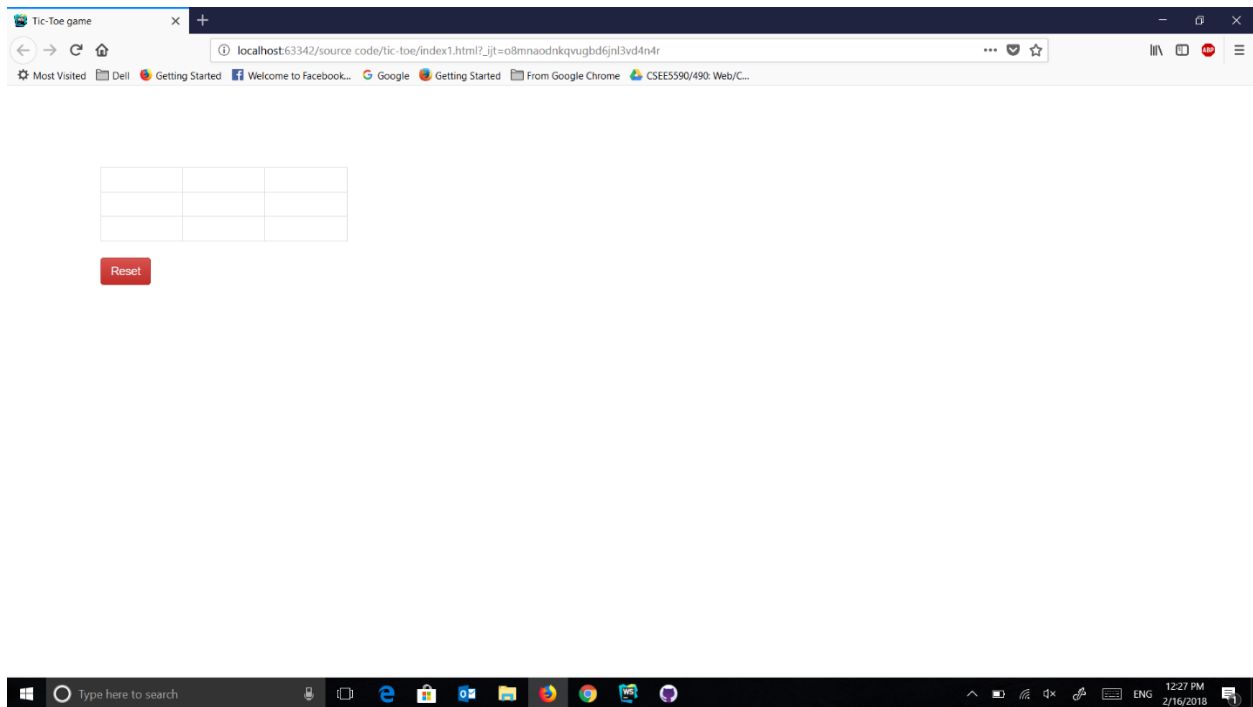
Lab Assignment #2

If we search for the topic and source

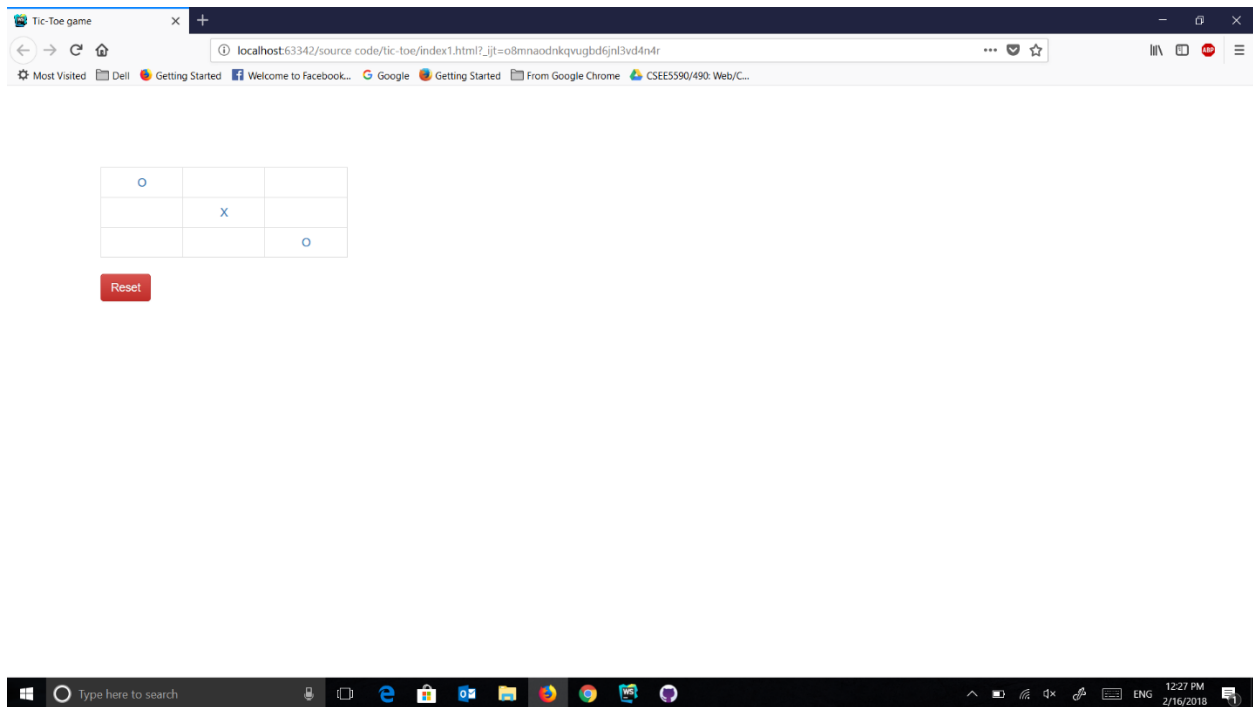


Lab Assignment #2

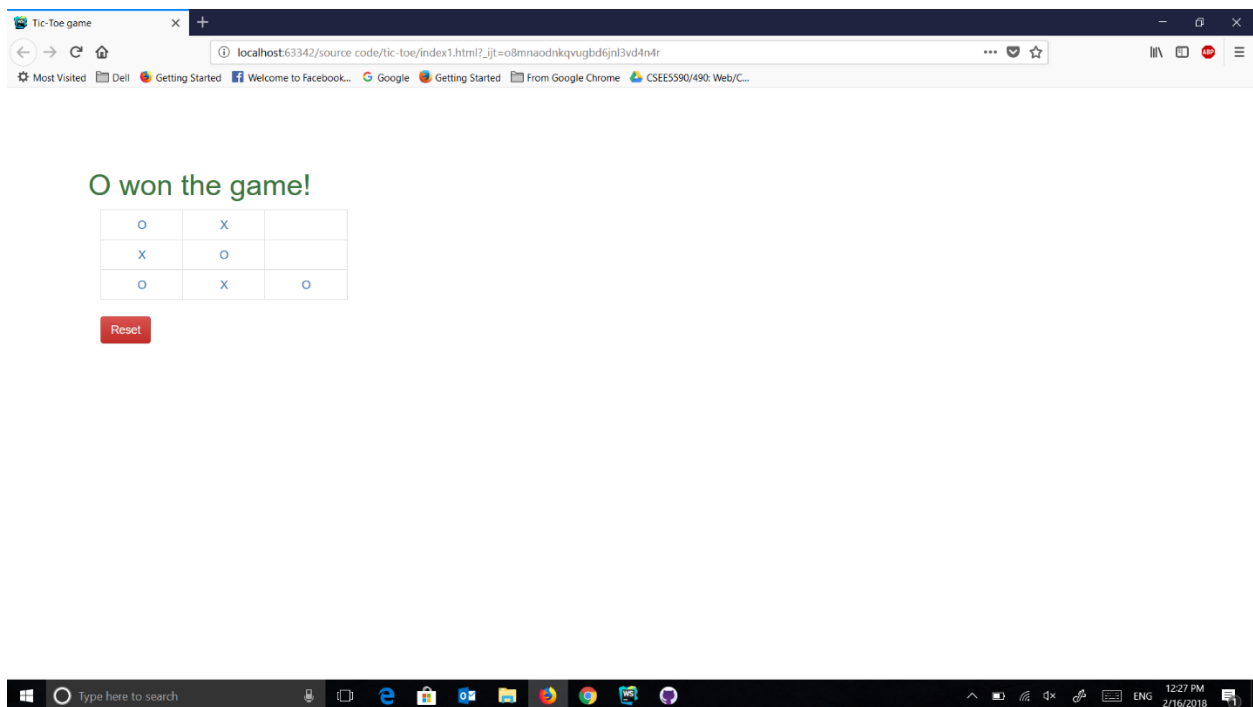
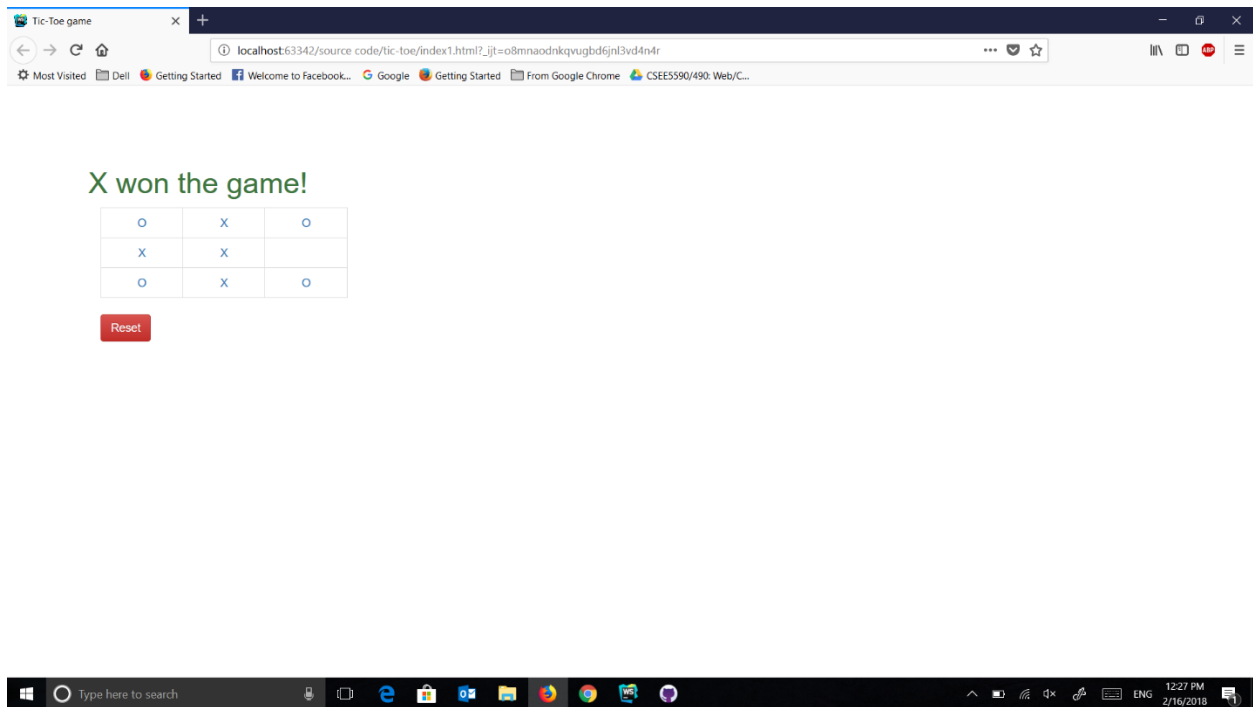
Part 2: Tie-Toe Game



the first player is distinguished by X and the second player by O



Lab Assignment #2



Lab Assignment #2

Code Snippet:

```
var loginid = localStorage.getItem("resid");
console.log(loginid);
$("#page").text("welcome " + loginid);

function response(source,topic) {

    var xhttp = new XMLHttpRequest();

    xhttp.open('GET',"https://newsapi.org/v2/everything?q="+topic+"&sources="+source+"&api
Key=b5468b9937314c37ae472c8d30430664", false);
    xhttp.send();
    return xhttp;
}

function showdata(news) {

    console.log(news.articles);
    $("#card1").show();
    $("#news1").text(news.articles[0].title);
    $("#news11").text(news.articles[0].description);
    $("#news111").attr("href",news.articles[0].url);
    $("#image1").attr("src",news.articles[0].urlToImage);
    $("#time1").text(news.articles[0].publishedAt);
    $("#card2").show();
    $("#news2").text(news.articles[1].title);
    $("#news21").text(news.articles[1].description);
    $("#news211").attr("href",news.articles[1].url);
    $("#image2").attr("src",news.articles[1].urlToImage);
    $("#time2").text(news.articles[1].publishedAt);
    $("#card3").show();
    $("#news3").text(news.articles[2].title);
    $("#news31").text(news.articles[2].description);
    $("#news311").attr("href",news.articles[2].url);
    $("#image3").attr("src",news.articles[2].urlToImage);
    $("#time3").text(news.articles[2].publishedAt);
    $("#card4").show();
    $("#news4").text(news.articles[3].title);
    $("#news41").text(news.articles[3].description);
    $("#news411").attr("href",news.articles[3].url);
    $("#image4").attr("src",news.articles[3].urlToImage);
    $("#time4").text(news.articles[3].publishedAt);
    $("#card5").show();
    $("#news5").text(news.articles[4].title);
    $("#news51").text(news.articles[4].description);
    $("#news511").attr("href",news.articles[4].url);
    $("#image5").attr("src",news.articles[4].urlToImage);
    $("#time5").text(news.articles[4].publishedAt);
    $("#card6").show();
    $("#news6").text(news.articles[5].title);
    $("#news61").text(news.articles[5].description);
    $("#news611").attr("href",news.articles[5].url);
    $("#image6").attr("src",news.articles[5].urlToImage);
    $("#time6").text(news.articles[5].publishedAt);
    $("#card7").show();
    $("#news7").text(news.articles[6].title);
    $("#news71").text(news.articles[6].description);
    $("#news711").attr("href",news.articles[6].url);
    $("#image7").attr("src",news.articles[6].urlToImage);
    $("#time7").text(news.articles[6].publishedAt);
    $("#card8").show();
}
```


Lab Assignment #2

```
$("#news8").text(news.articles[7].title);
$("#news81").text(news.articles[7].description);
$("#news811").attr("href",news.articles[7].url);
$("#image8").attr("src",news.articles[7].urlToImage);
$("#time8").text(news.articles[7].publishedAt);
}

$(document).ready(function () {

    $("#search").click(function () {

        topic = $("#topic").val();
        source= $("#source").val();
        console.log(source);
        console.log(topic);
        resp = response(source,topic);
        if (resp.status == 200){
            showdata(JSON.parse(resp.responseText));
            console.log("success");
        }else{
            console.log("not find");
        }
    });
});
```

Tic-Toe Game:

```
var app = angular.module('app1', ['ng']);

(function() {
    var Board1 = (function(){
        var SIZE1, NOUGHT1, PLAYER_MARKERS1,EMPTY1, Cell1, CROSS1;

        SIZE1 = 3;
        CROSS1 = 'X';
        EMPTY1 = ' ';
        NOUGHT1 = 'O';

        PLAYER_MARKERS1 = [NOUGHT1, CROSS1];

        Cell1 = (function() {
            function Cell1(marker){
                this.marker1 = marker;
                this.winning1 = false;
            }
            Cell1.prototype.mark1 = function(marker) {
                this.marker = marker1;
            }
            Cell1.prototype.hasBeenPlayed1 = function() {
                return this.marker != EMPTY;
            }
            return Cell1;
        })();

        function Board1() {
            this.reset1();
        }

        Board1.prototype.reset = function(){
            this.currentPlayer1 = 0;
            this.grid1 = [];
            this.won1 = false;
            this.gameGoingOn1 = true;
        }
    })();
});
```

Lab Assignment #2

```
for(var i=0; i < SIZE1; i++) {
    var row1 = [];

    for(var j=0; j < SIZE1; j++) {
        row1.push(new Cell1(EMPTY));
    }

    this.grid.push(row1);
}

return this.grid1;
};

Board.prototype._checkRow = function(rowIndex) {
    var numberOfNoughts1 = 0;
    var numberOfCrosses1 = 0;

    for(var columnIndex = 0; columnIndex < SIZE1; columnIndex++) {
        var cell1 = this.grid1[rowIndex];
        cell1 = cell1[columnIndex];
        var cellMarker1 = cell1.marker;
        if(cellMarker1 == EMPTY) {
            return false;
        }
        if(cellMarker1 == NOUGHT1) {
            numberOfNoughts1++;
        } else if(cellMarker1 == CROSS1) {
            numberOfCrosses1++;
        }
    }

    if(numberOfNoughts1 == SIZE) {
        return NOUGHT1;
    } else if(numberOfCrosses1 == SIZE) {
        return CROSS1;
    }
};

Board.prototype._checkColumn = function(columnIndex) {
    var numberOfNoughts1 = 0;
    var numberOfCrosses1 = 0;

    for(var rowIndex1 = 0; rowIndex < SIZE; rowIndex++) {
        var cell1 = this.grid[rowIndex][columnIndex];
        var cellMarker1 = cell.marker;
        if(cellMarker1 == EMPTY) {
            return false;
        }
        if(cellMarker1 == NOUGHT) {
            numberOfNoughts1++;
        } else if(cellMarker1 == CROSS) {
            numberOfCrosses1++;
        }
    }

    if(numberOfNoughts1 == SIZE) {
        return NOUGHT1;
    } else if(numberOfCrosses1 == SIZE) {
        return CROSS1;
    }
}
```

Lab Assignment #2

```
Board.prototype._checkDiagonal1 = function() {
    var numberOfNoughts1 = 0;
    var numberOfCrosses1 = 0;

    for(var i = 0; i<SIZE; i++) {
        var cellMarker1 = this.grid1[i][i].marker;
        if(cellMarker1 == EMPTY) {
            return false;
        }
        if(cellMarker1 == NOUGHT) {
            numberOfNoughts1++;
        } else if(cellMarker1 == CROSS) {
            numberOfCrosses1++;
        }
    }

    if(numberOfNoughts1== SIZE) {
        return NOUGHT1;
    } else if(numberOfCrosses1 == SIZE) {
        return CROSS1;
    }
}

Board.prototype._checkDiagonal2 = function() {
    var numberOfNoughts1 = 0;
    var numberOfCrosses1 = 0;

    for(var i = 0; i<SIZE1; i++) {
        var cellMarker1 = this.grid1[i][SIZE-i-1].marker;
        if(cellMarker1 == EMPTY) {
            return false;
        }
        if(cellMarker1 == NOUGHT) {
            numberOfNoughts1++;
        } else if(cellMarker1 == CROSS) {
            numberOfCrosses1++;
        }
    }

    if(numberOfNoughts1 == SIZE) {
        return NOUGHT1;
    } else if(numberOfCrosses1 == SIZE) {
        return CROSS1;
    }
}

Board.prototype._setWinner = function(marker) {
    this.won1 = true;
    this.winning_marker1 = marker;
    this.gameGoingOn1 = false;
};

Board.prototype._markWinnerRow = function(rowIndex) {
    for(var i = 0; i<SIZE1; i++) {
        this.grid[rowIndex][i].winning1 = true;
    }
}

Board.prototype._markWinnerColumn = function(columnIndex) {
    for(var i = 0; i<SIZE1; i++) {
        this.grid[i][columnIndex].winning1 = true;
    }
}
```

Lab Assignment #2

```
Board.prototype._markWinnerDiagonal1 = function() {
    for(var i = 0; i<SIZE1; i++) {
        this.grid1[i][i].winning1 = true;
    }
}

Board.prototype._markWinnerDiagonal2 = function() {
    for(var i = 0; i<SIZE1; i++) {
        this.grid[i][SIZE1-i-1].winning1 = true;
    }
}

Board.prototype.checkWinner = function() {
    var rowIndex1 = 0;
    var columnIndex1 = 0;
    var diagonal1 = 0;
    var diagonal2 = 0;

    for(rowIndex=0; rowIndex < SIZE1; rowIndex++) {
        var val1 = this._checkRow(rowIndex1);
        if(val1) {
            this._setWinner1(val1);
            this._markWinnerRow1(rowIndex1);
        }
    }

    for(columnIndex = 0; columnIndex < SIZE1; columnIndex++) {
        val2 = this._checkColumn(columnIndex);
        if(val2) {
            this._setWinner(val2);
            this._markWinnerColumn(columnIndex);
        }
    }

    val3 = this._checkDiagonal1();
    if(val3) {
        this._setWinner(val3);
        this._markWinnerDiagonal3();
    }

    val4 = this._checkDiagonal2();
    if(val4) {
        this._setWinner(val4);
        this._markWinnerDiagonal2();
    }
};

Board.prototype.playCell = function(cell) {
    if(!(this.gameGoingOn)) {
        return;
    }
    if(cell.hasBeenPlayed()) {
        return;
    }
    cell.mark1(this.currentPlayerMarker1());
    this.checkWinner1();
    this.switchPlayer1();
};

Board.prototype.currentPlayerMarker = function() {
```

Lab Assignment #2

```
        return PLAYER_MARKERS[this.currentPlayer];
    }

    Board.prototype.switchPlayer = function() {
        this.currentPlayer = 1 - this.currentPlayer;
    }

    return Board1;
})();

angular.module('app1').factory('Board1', function(){
    return Board1;
});

}).call(this);

app1.controller('BoardCtrl1', function($scope, Board){
    $scope.board = new Board1;
});
```