A valid Cordova project must contain a `myCordovaProjectSolution\www` folder.

All of the **HTML, CSS and JavaScript** edits required must take place in the above-outlined folder, as per:

```
myCordovaProjectSolution\www\index.html //To do HTML edits here

myCordovaProjectSolution\www\js\index.js //To do JavaScript edits here

myCordovaProjectSolution\www\css\index.css //To do CSS edits here

myCordovaProjectSolution\www\img //To attach any image edits here
```

In the upcoming pages you should find 4 different prototype scenarios outlined and accompanied by their provided solutions and APKs

Example 01: REST API integration

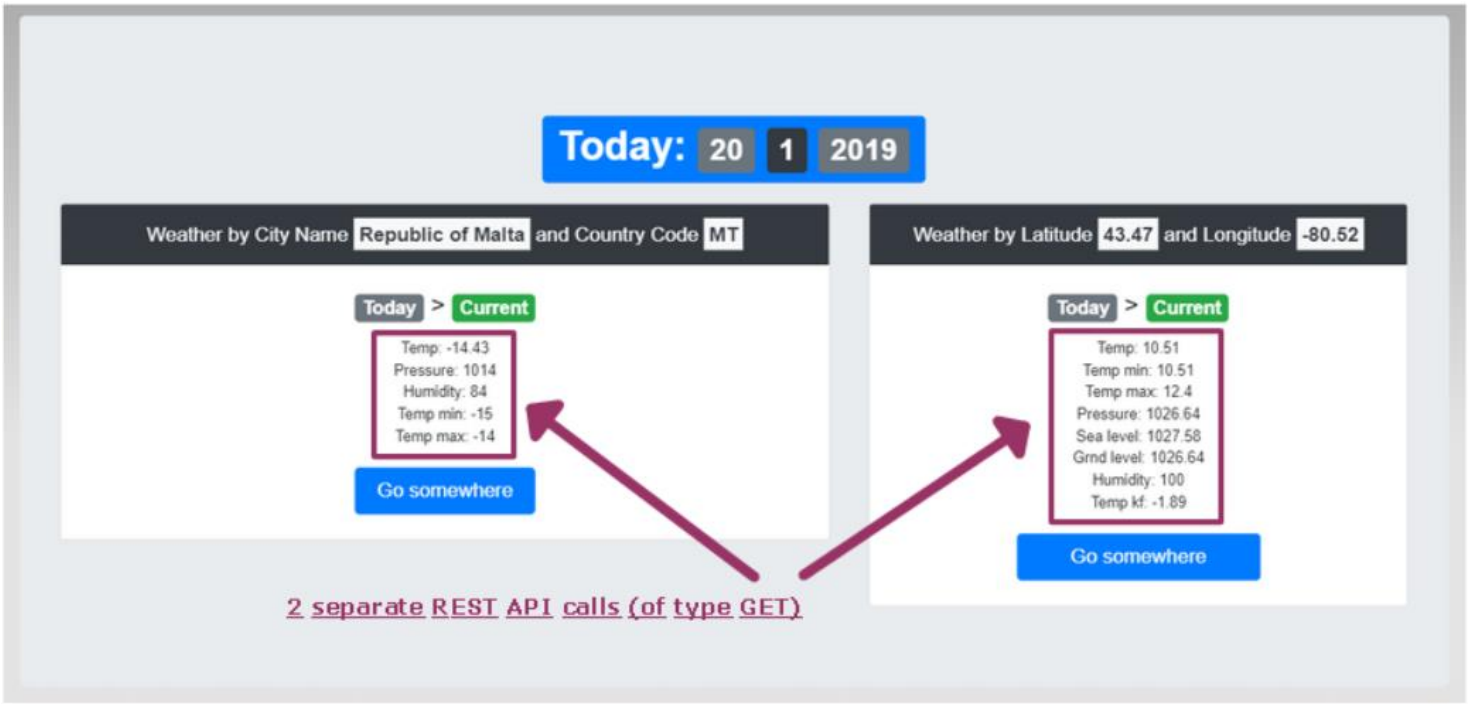
Access the following solution: **myWeatherApp** *(found inside Examples folder)*

Please run solution from: **cordova-simulate plugin**

```
self.pluginMethods = {
    weather: {
        getWeatherByLatAndLong: function(latitude, longitude) {
            // Get a free key at http://openweathermap.org/. Replace the "Your_Key_Here" string with that key.
            var _latitude = latitude || 35.9167; //defaults to current country, ie. Malta
            var _longitude = longitude || 14.4333; //defaults to current country, ie. Malta

            var OpenWeatherReq_REST = "http://api.openweathermap.org/data/2.5/weather?lat=" + _latitude + "&lon=" + _longitude +
            "&appid=" + self.OpenWeatherAppKey + "&units=metric";

            _getWeather(OpenWeatherReq_REST);
        },
        getWeatherByCityAndCountry: function (cityName, countryCode) {
            var _cityName = cityName || "Republic%20of%20Malta"; //If not provided, a default value is set
            var _countryCode = countryCode || "MT"; //If not provided, a default value is set

            var OpenWeatherReq_REST = "http://api.openweathermap.org/data/2.5/forecast?q=" + _cityName + "," + _countryCode +
            "&mode=json&units=metric&appid=" + self.OpenWeatherAppKey;

            _getWeather(OpenWeatherReq_REST, { cityName: _cityName, countryCode: _countryCode });
        }
};
```

2 typical methods used in the provided example

Screenshot taken from: ***myWeatherApp\www\index.js***



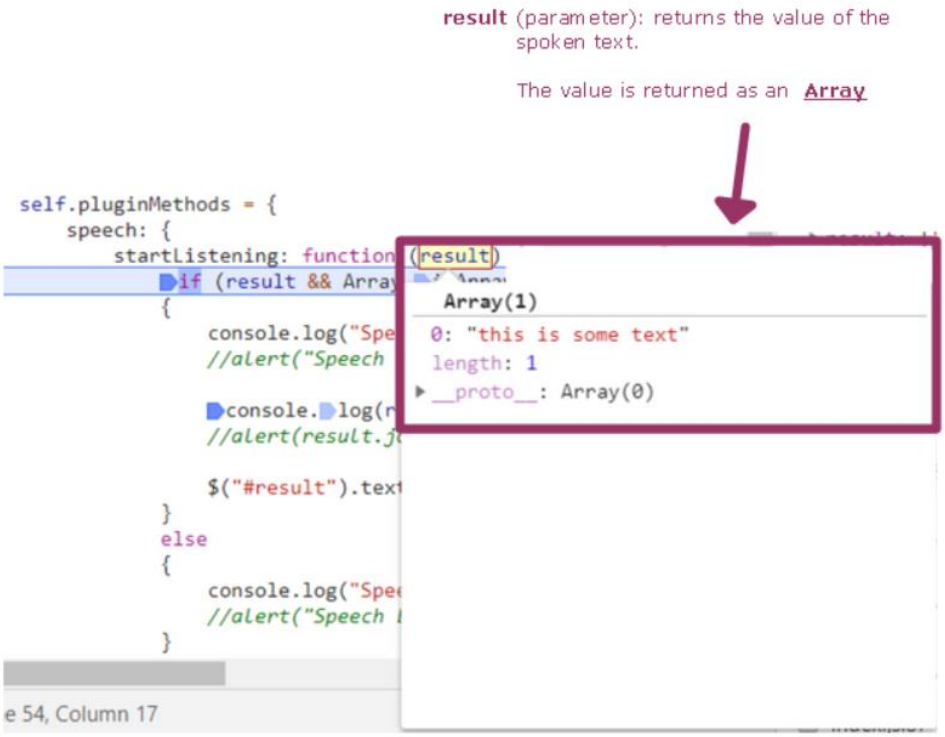
Result output of: ***myWeatherApp*** *solution*

```
//IMPORTANT NOTE ON REST Requests
    //In your assignment scenario you may need to trigger requests via $.ajax method, as per below.

    $.ajax({
      type: "", // GET or POST or PUT or DELETE
      url: "", // URL path to call
      data: { // Data to pass in REST API call (please follow options outlined in the REST service specifications)
      },
      contentType: "application/json;", //Format expected by the Server
      dataType: "json", //Format expected by the Client in response
      success: function(data) { //Callback to be triggered should the request be successful
      },
      error: function(data) { //Callback to be triggered should the request fail
      }
    });

    // IMPORTANT: Make sure to include jquery (full version) to be able to access $.ajax \ $.getJSON methods:

    /*
    <script src="https://code.jquery.com/jquery-3.3.1.min.js"
          integrity="sha256-FgpCb/KJQlLNfOu91ta32o/NMZxltwRo8QtmkMRdAu8="
          crossorigin="anonymous"></script>
    */
```

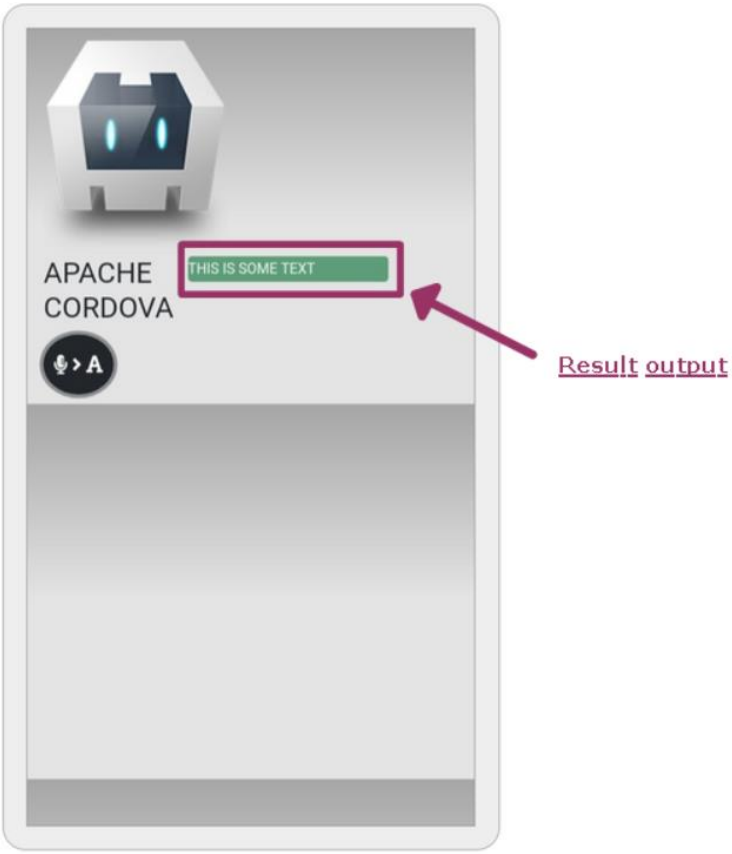
Example 02: Speech-to-text plugin integration

Access the following solution: **voiceToTextConverterApp** *(found inside Examples folder)*

Please run solution from: **android emulator**



Screenshots taken from: ***voiceToTextConverterApp\www\index.js***



Result output of: ***voiceToTextConverterApp*** *solution*

Example 03: File plugin integration

Access the following solution: **fileReadWriteApp** *(found inside Examples folder)*

Can run solution from: **android emulator \ cordova-simulate plugin**

```
readFile: function (textfile_name) {
    window.requestFileSystem(LocalFileSystem.PERSISTENT, 0, function (fs) {
        fs.root.getFile(textfile_name + ".txt", { create: false }, function (fileEntry) {
            fileEntry.file(function (file) {
                //On read file - SUCCESS
                var reader = new FileReader();

                reader.onloadend = function() {
                    var textToPrint = "Successful file read \n\n";

                    var result = this.result.trim();

                    if (result == "")
                        textToPrint += "No data found inside the \'" + textfile_name + "\' text file";
                    else {

                        //Displaying all text content (from text file) inside an alert box
                        textToPrint += "The following text was inside the \'" + textfile_name + "\' text file: \n\n" + result;
                    }

                    alert(textToPrint);
                };

                reader.readAsText(file); }, function (result) { /*On read file - ERROR*/ });
    }, function (result) {
```

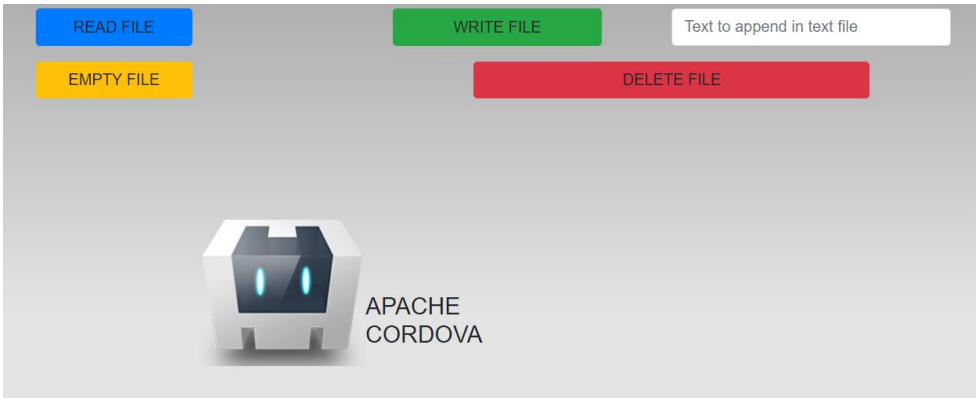
Reading all text content from text file

Printing read content

Screenshots taken from: ***fileReadWriteApp\www\index.js***

```
writeFile: function (textfile_name, textToWrite, isAppend, clearAllText, readFile) {
    isAppend = (typeof isAppend == "boolean") ? isAppend : true;
    readFile = (typeof readFile == "boolean") ? readFile : false;
    clearAllText = (typeof clearAllText == "boolean") ? clearAllText : false;

    window.requestFileSystem(LocalFileSystem.PERSISTENT, 0, function (fs) {
        fs.root.getFile(textfile_name + ".txt", { create: true, exclusive: false },

            fileEntry.createWriter(function (fileWriter) {

                fileWriter.onwriteend = function() { ... };

                fileWriter.onerror = function (e) { ... };

                // If we are appending data to file, go to the end of the file.
                if (isAppend) { ... }

                if (clearAllText)
                    fileWriter.truncate(0); //Clearing all data
                else {
                    //Adding text content inside text file
                    fileWriter.write(textToWrite + " ");
                }
            });
```

Writing content



Result output of: ***fileReadWriteApp*** *solution*

Example 04: Media plugin integration

Access the following solution: **audioFileCreator-PlayerApp** *(found inside Examples folder)*

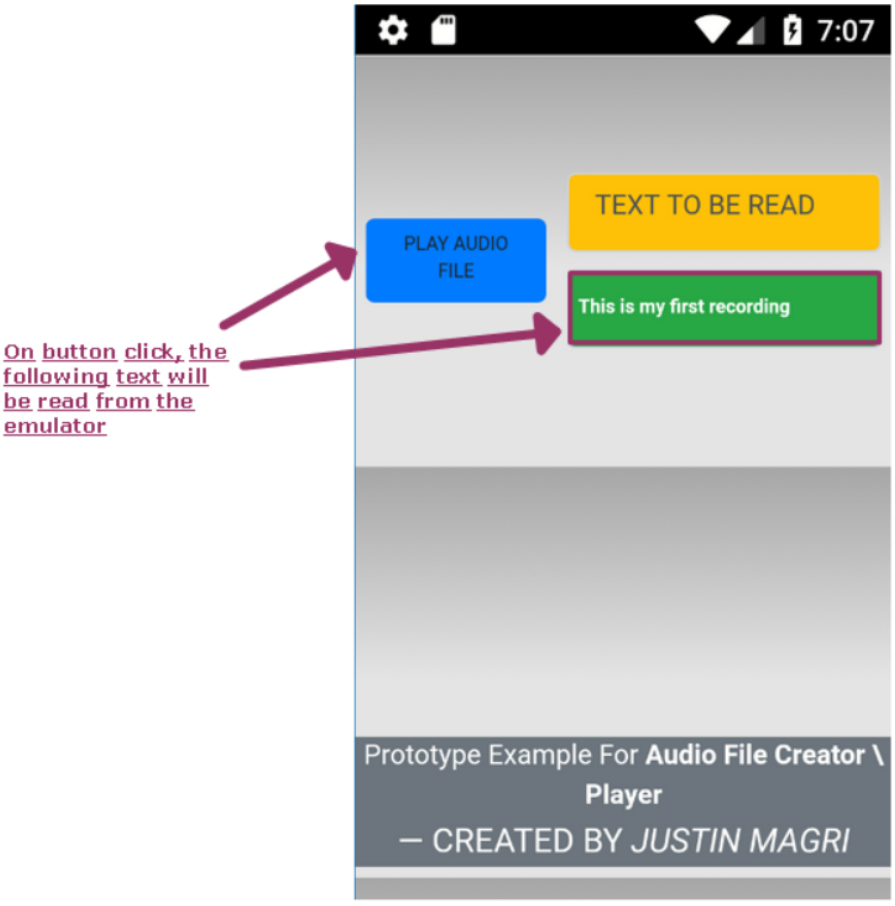
Please run solution from: **android emulator**

```
//Add this method in your assignment - DO NOT AMEND
function playAudio(fileObj) {
    var _folderPath = fileObj.localURL;

    var media = new Media(_folderPath, function (successResult) {
        //On successful media play\stop\record action, do something (optionally)
        customOnAfterMediaActionSuccess(_folderPath, successResult); //Not required in your case
    }, function (errorResult) {
        //On failed media play\stop\record action, do something (optionally)
    });

    media.play(); //This will literally play the audio on your smartphone\emulator

//Add this method in your assignment - DO NOT AMEND
function b64toBlob(b64Data, contentType, sliceSize) { ... }

//Add this method in your assignment - DO NOT AMEND
//Only this method needs to be triggered in your assignment
function saveBase64AsAudioFile(folderpath, filename, b64Data, contentType){ ... }
```

Trigger this method to play the audio file that was previously created

This procedure will create an audio file in the provided directory path

Screenshots taken from: ***audioFileCreator-PlayerApp\www\index.js***



Result output of: ***audioFileCreator-PlayerApp*** *solution*