

Mobile User Experience 6MMCS001W

Coursework 2 Report - Mobile App Implementation

Author: Shenal Anthony
Student ID: 2018383/w1742306

Group Number: 30

Links to Website:

URL to Video: <https://www.youtube.com/watch?v=ayaLxSSBB9I>

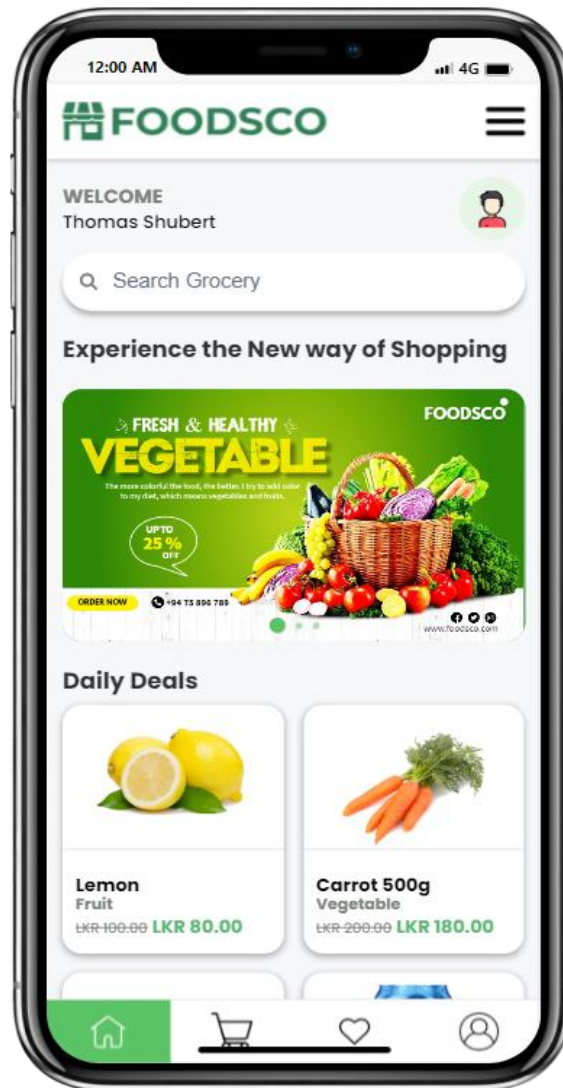
Content

Content	2
1. Design and implementation of mobile web app features	3
2.1 Eye catching introduction	3
2.2 Incorporation of a short video	4
2.3 POI	5
2.5 Integration of google maps	6
2.5 List creation & sorting and filtering widgets	7
2.6 Booking form	14
2.7 Favourites list	19
2.8 Gamification (might need to be combined with 2.9)	24
2.9 System feedback	25
2.10 Scores & points gained (might need to be combined with 2.7)	28
2.11 Comments	29
2.13 AR/QR	36
2.14 Navigation	37
2.15 Differentiation between iPhone and iPad implementation	40
References	41

1. Design and implementation of mobile web app features

In the sections below provide and explain the proposed design and the code you wrote to create the following main features and functionality of the mobile web app.

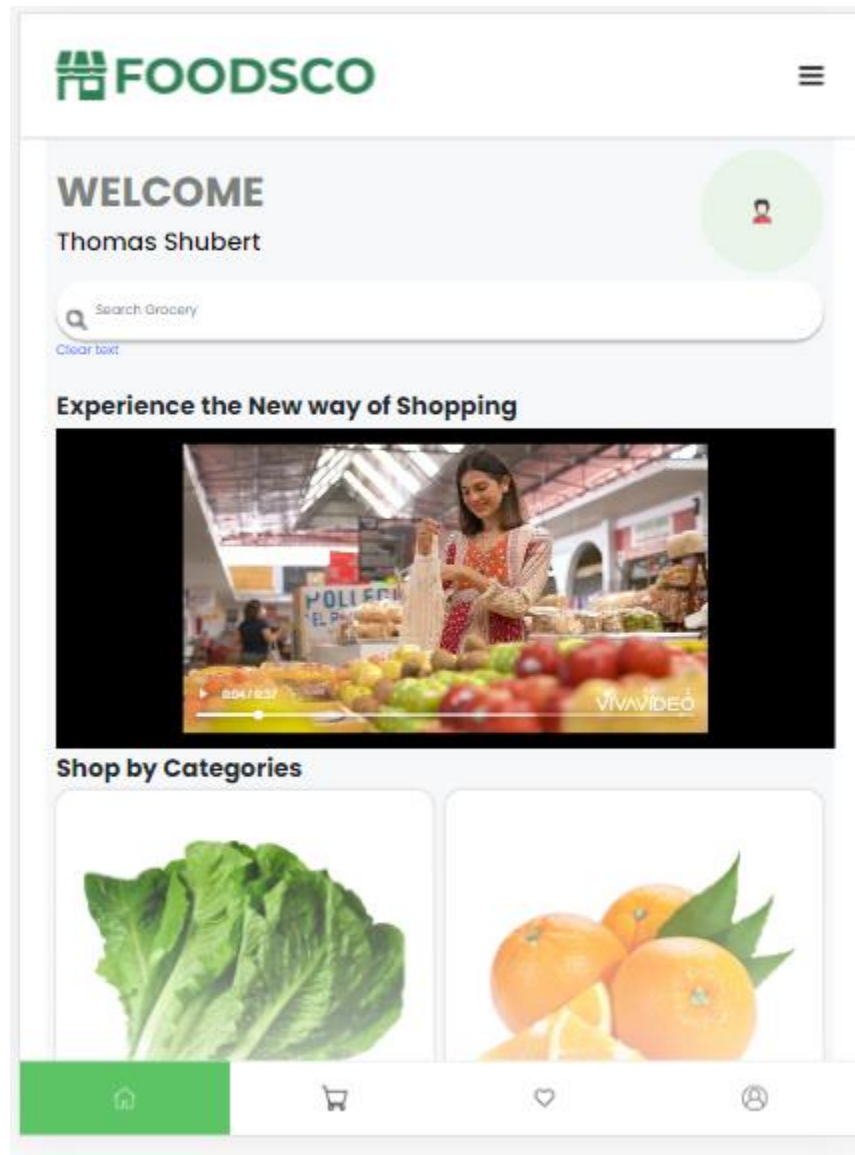
2.1 Eye catching introduction



An eye-catching image slider has been added on the home page with introductory information and promotion details.

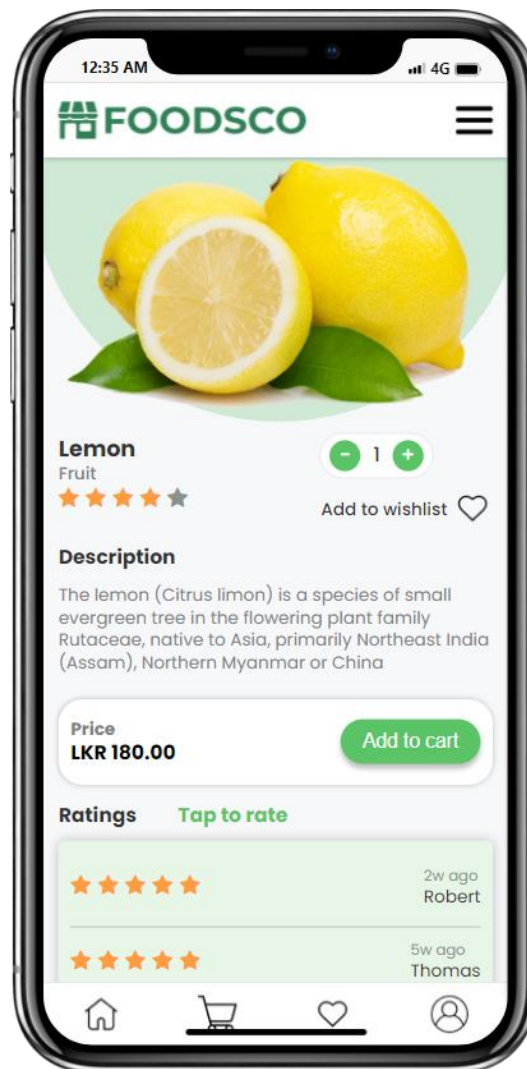
2.2 Incorporation of a short video

We have incorporated a short video on our home page which is showing our supermarket. This helps the user to get a good first impression. The video consists of the prototype design demo. We have used the jQuery mobile iframe to insert the video to the site.



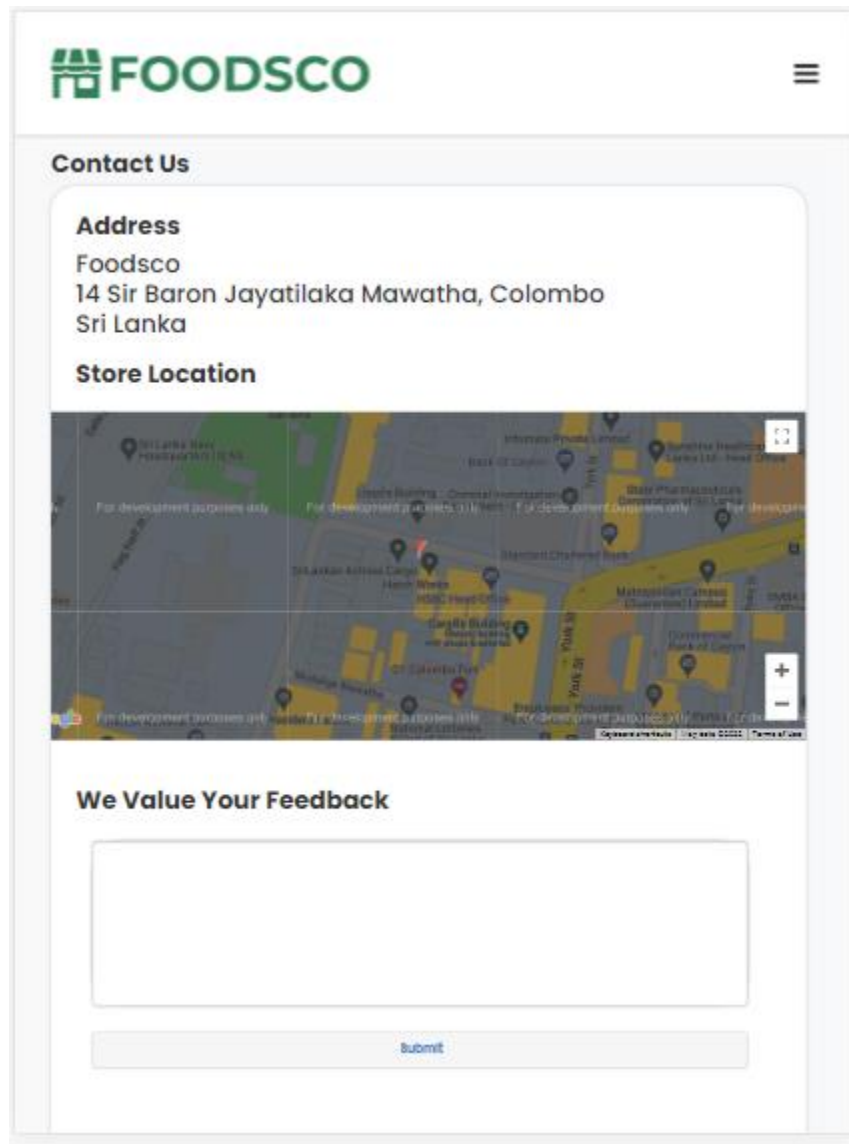
2.3 POI

Point of interest can be very important in attracting users. Our application used a variety of techniques to keep the user's favorite features appealing. To ensure quality design, visual balancing and emphasis classes were used in the design process. When it came to icon and button placement, we focused on using them in prominent places on the page, such as the center and corners. In addition, when transitioning to iPhone and iPad views, we used different sizes for elements such as buttons to maintain the application's dynamic nature.



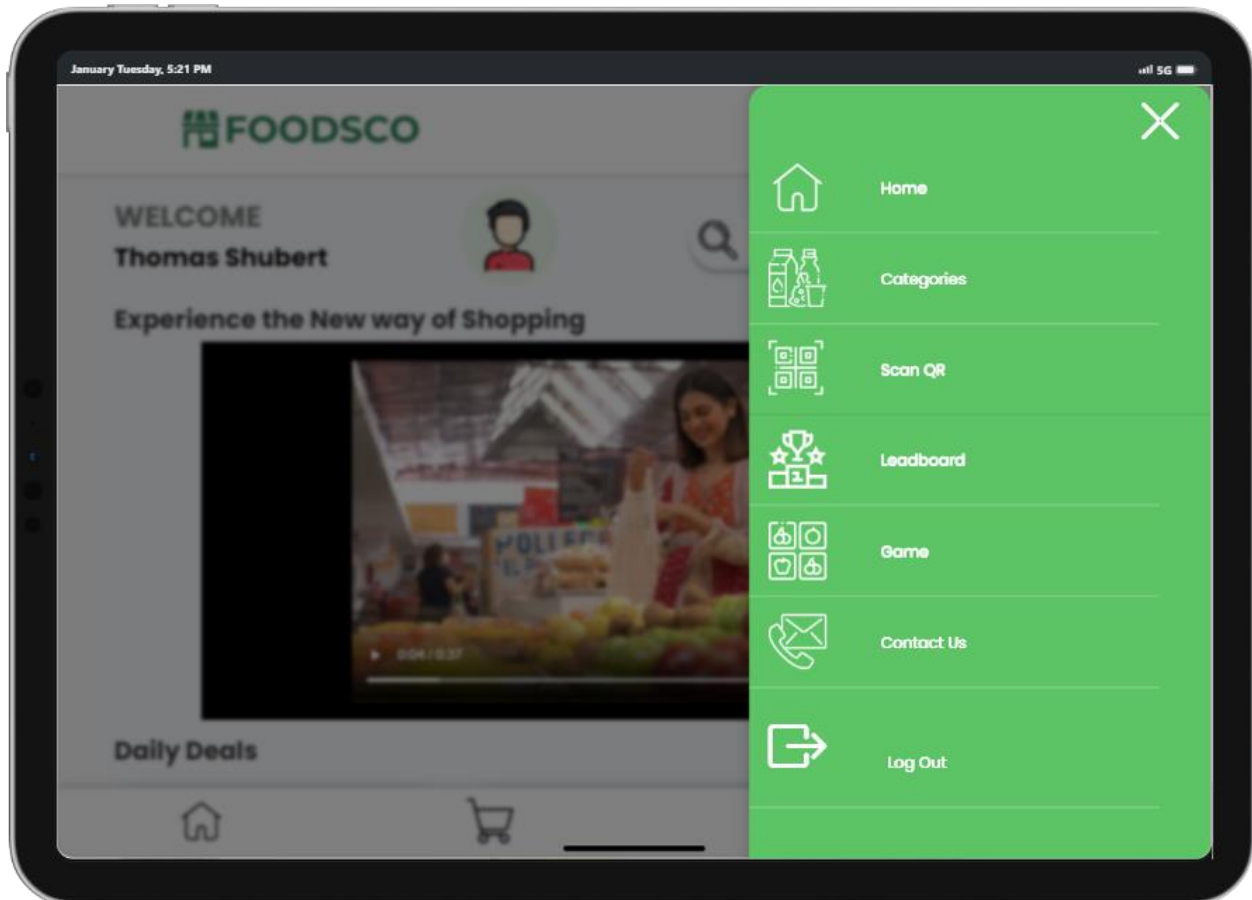
2.5 Integration of google maps

We used the map on our Contact Us page to show where our store is located. To integrate the maps, we used the Google Maps API from the Google Cloud Console. The view of the maps on the iPad and iPhone is nearly identical, with the only difference being the size of the container used to hold the map.

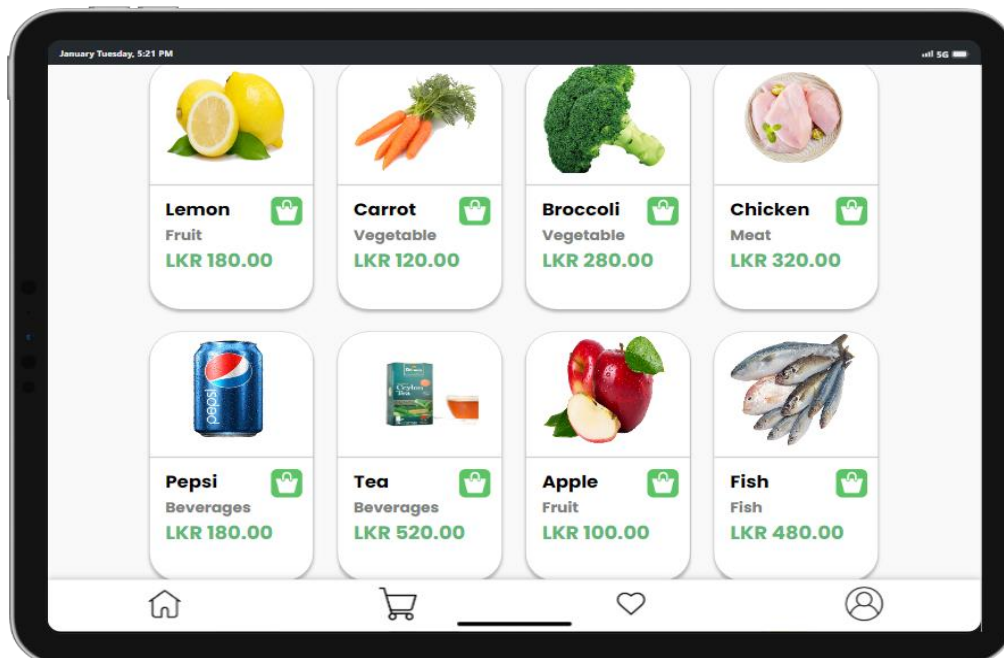
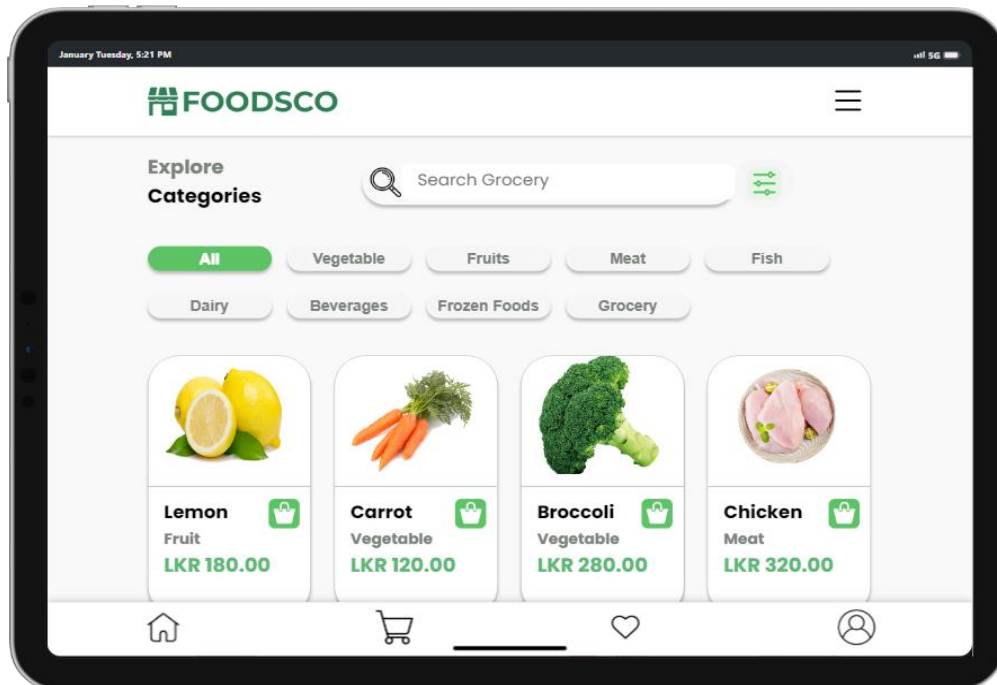


2.5 List creation & sorting and filtering widgets

In the menu the categories link is where the user can access the list of POI's. The category page can be access from the daily deals and also from shop by category section.



The list of POI's are displayed as shown below image.




```

26
27 function documentReady() {
28     $.getJSON("products.json", function (data) {
29         var product_data = "";
30
31         $.each(data, function (key, value) {
32             product_data +=
33                 '<div class="product" id="product' +
34                 value.prod_id +
35                 '>' +
36                 '<a href="SingleProduct.html?prod_id=' +
37                 value.prod_id +
38                 '" rel="external">' +
39                 '' +
42                 "</a>" +
43                 '<hr class="hr">' +
44                 '<div class="row" id="prod_descr_area">' +
45                 '<div class="col-sm-8" id="">' +
46                 '<div class="prod_name" id="prod_name">' +
47                 '<a href="SingleProduct.html?prod_id=' +
48                 value.prod_id +
49                 '" rel="external">' +
50                 value.prod_name +
51                 "</a>" +
52                 "</div>" +
53                 "</div>" +
54                 '<div class="col-sm-3" id="">' +
55                 '' +
56                 "</div>" +
57                 "</div>" +
58                 '<div class="prod_category" id="prod_category">' +
59                 value.prod_category +
60                 "</div>" +
61                 '<div class="prod_price" id="prod_price">LKR ' +
62                 value.prod_price +
63                 "</div>" +
64                 "</div>";
65         });
66         $("#product_row").html(product_data);
67     });
68 }

```

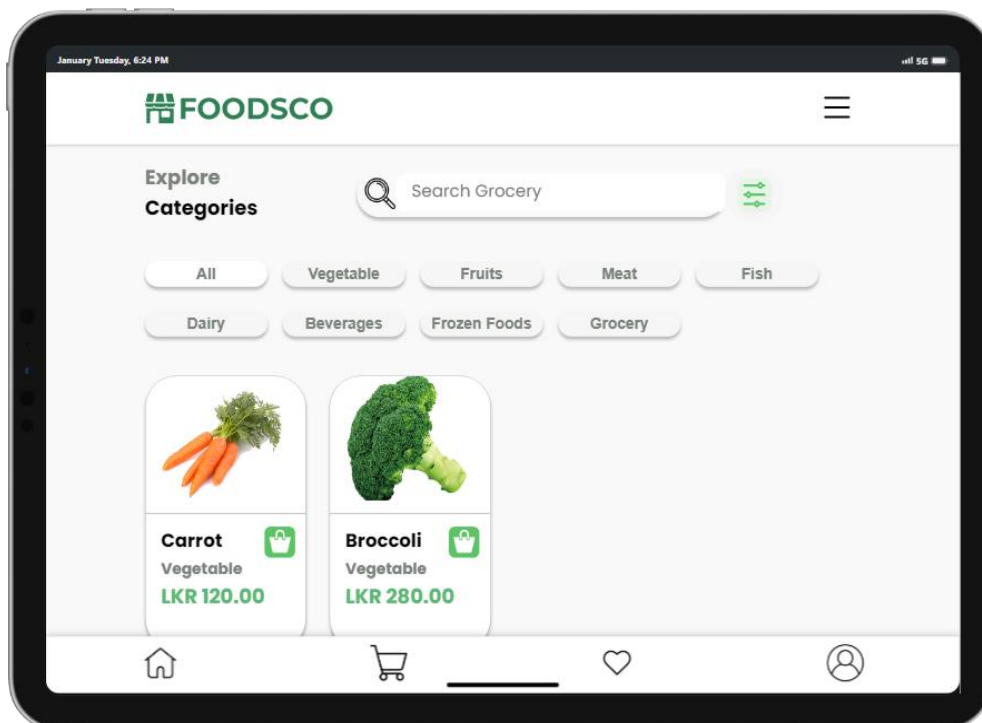
The detail of each POI is stored in a JSON file, so it is easy to access from any file and easy to perform the sorting and the filtering part.

```

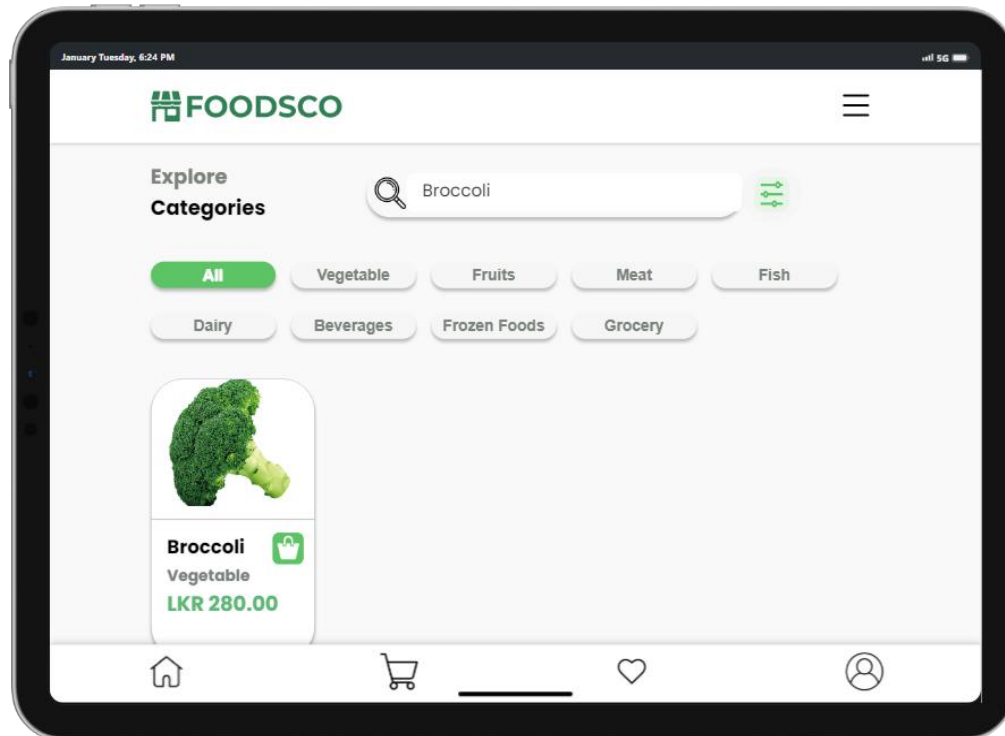
{
  "prod_id": "1",
  "prod_name": "Lemon",
  "prod_category": "Fruit",
  "prod_price": "180.00",
  "prod_img": "lemon.png",
  "prod_rating": 4,
  "prod_description": "The lemon (Citrus limon) is a species of small evergreen tree in the flowering plant family Rutaceae, native to Asia, primarily Northeast India (Assam)",
  "prod_comment": [
    {
      "name": "Thomas",
      "message": "Great fruit and fresh and very tasty and clean hope to buy it again",
      "date": "12/08/2021"
    },
    {
      "name": "Stewart",
      "message": "well packed and fully satisfied with this order 🍋.The packing was neat and the chocolates were delicious 🍫 And also fast delivery 🚚 th",
      "date": "15/08/2021"
    },
    {
      "name": "Tyler",
      "message": "Fast Delivery. It's good. Keep it up 🙌",
      "date": "20/08/2021"
    }
  ]
},
{
  "prod_id": "2",
  "prod_name": "Carrot",
  "prod_category": "Vegetable",
  "prod_price": "120.00",
  "prod_img": "carrot.png",
  "prod_rating": 5,
  "prod_description": "The carrot (Daucus carota subsp. sativus) is a root vegetable, typically orange in color, though purple, black, red, white, and yellow cultivars exist.",
  "prod_comment": [
    {
      "name": "Thomas",
      "message": "Great fruit and fresh and very tasty and clean hope to buy it again",
      "date": "12/08/2021"
    },
    {
      "name": "Stewart",
      "message": "well packed and fully satisfied with this order 🍋.The packing was neat and the chocolates were delicious 🍫 And also fast delivery 🚚 th",
      "date": "15/08/2021"
    },
    {
      "name": "Tyler",
      "message": "Fast Delivery. It's good. Keep it up 🙌",
      "date": "20/08/2021"
    }
  ]
}

```

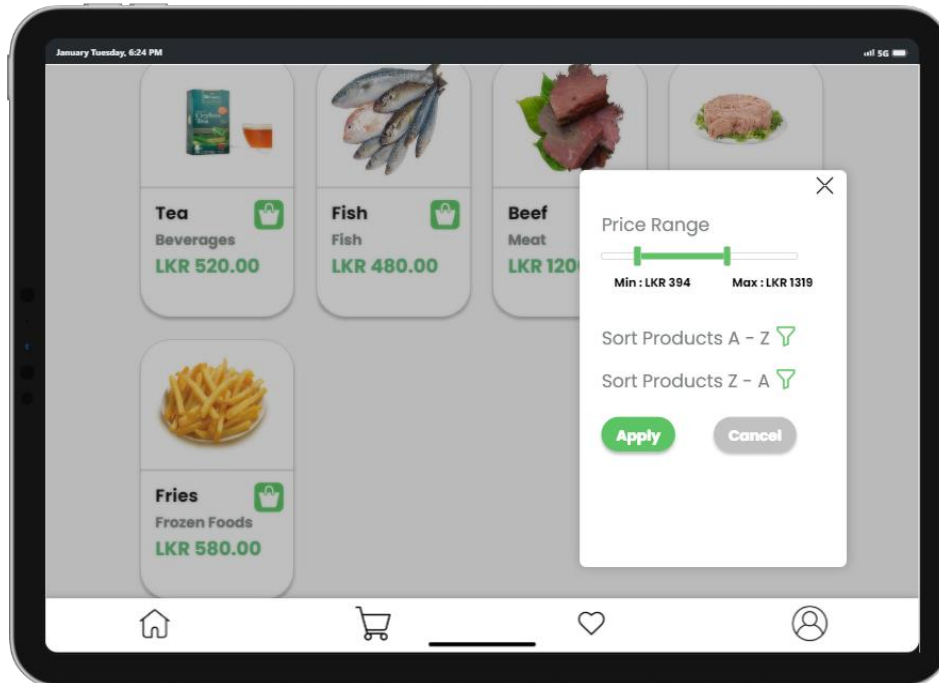
Both sorting and filtering are available to list the POI's in order. For the filtering multiple filters have been used. First on the filtering the category filtering the POI's are filter according to the products



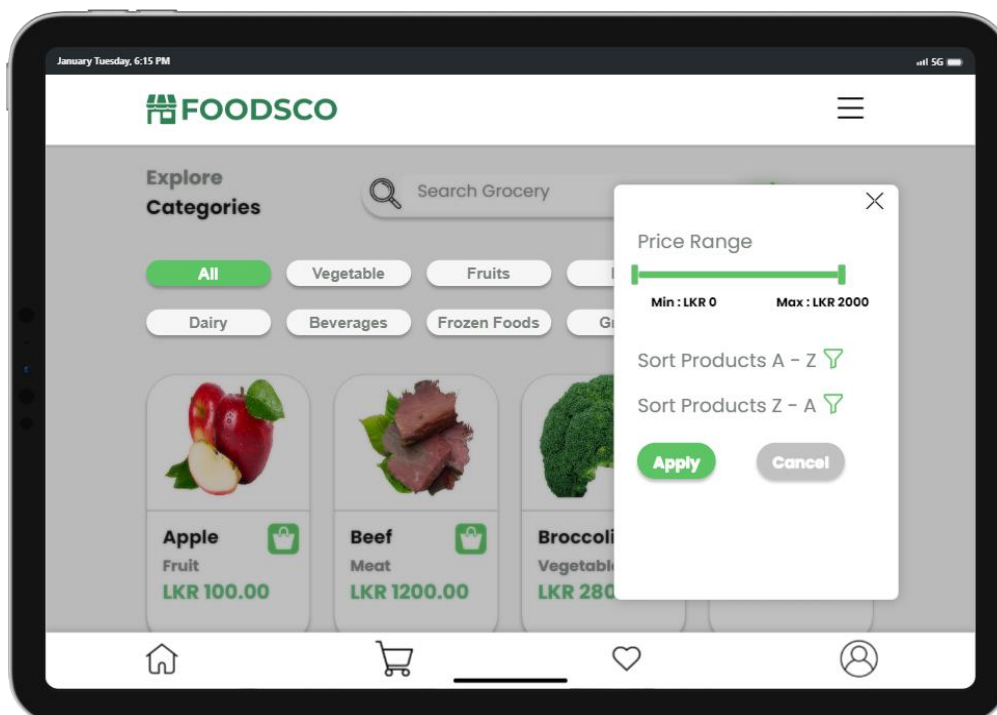
There is a search bar to sort a particular product and get that data.



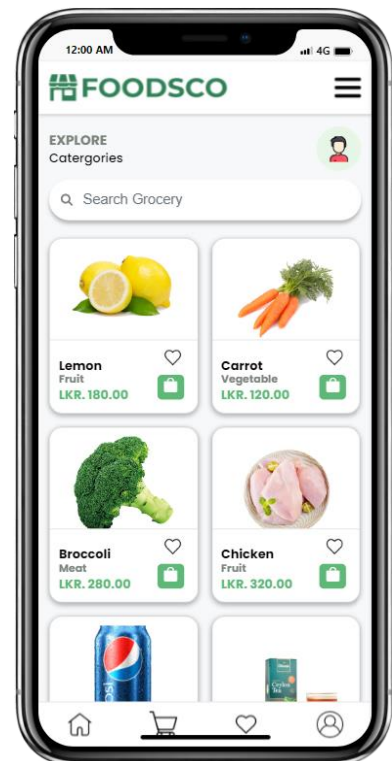
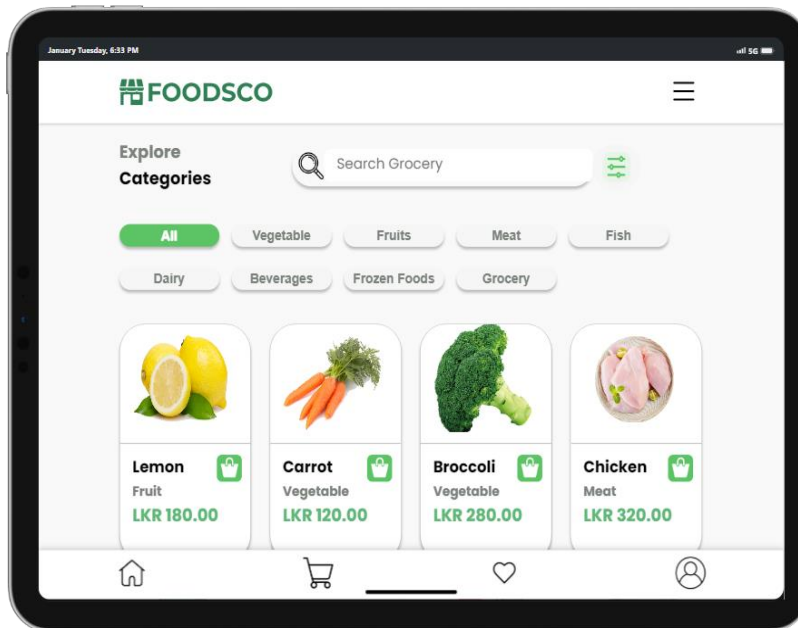
The next filter is price range filter to get the poi's in the selected price range.



The alphabetical sorting is used for sorting the POI's. Both Ascending and descending of the sorting are available.

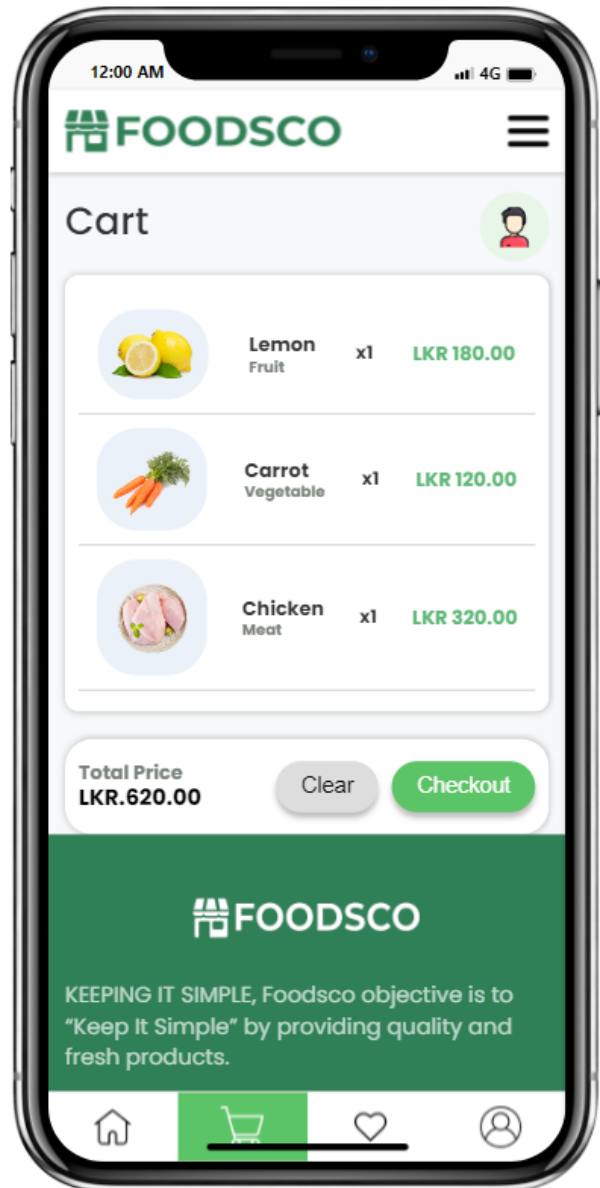
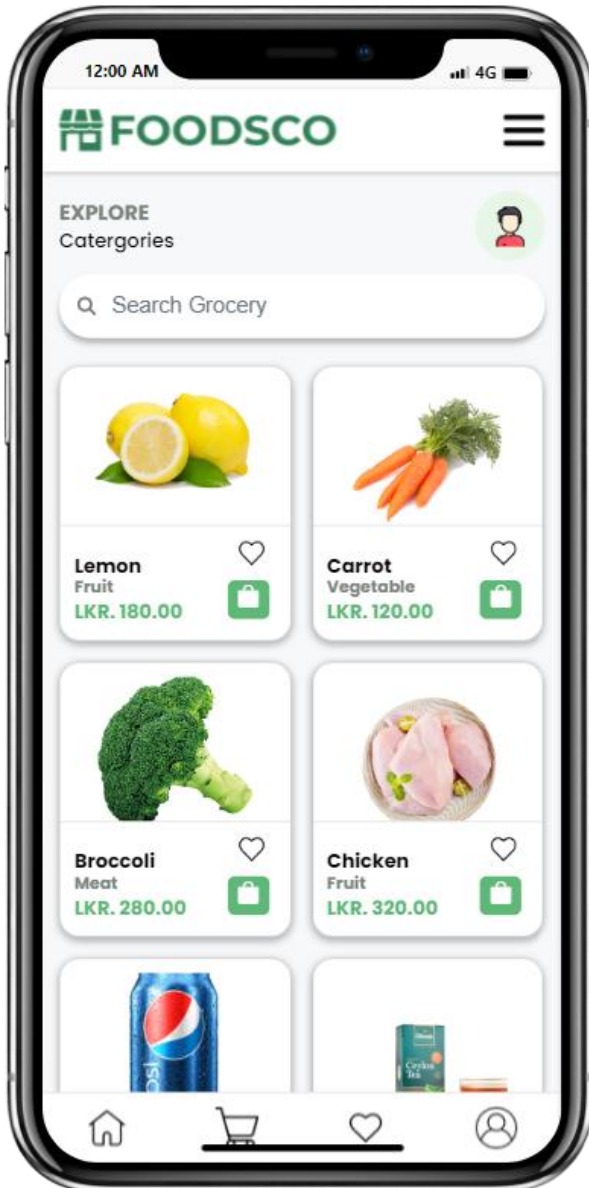


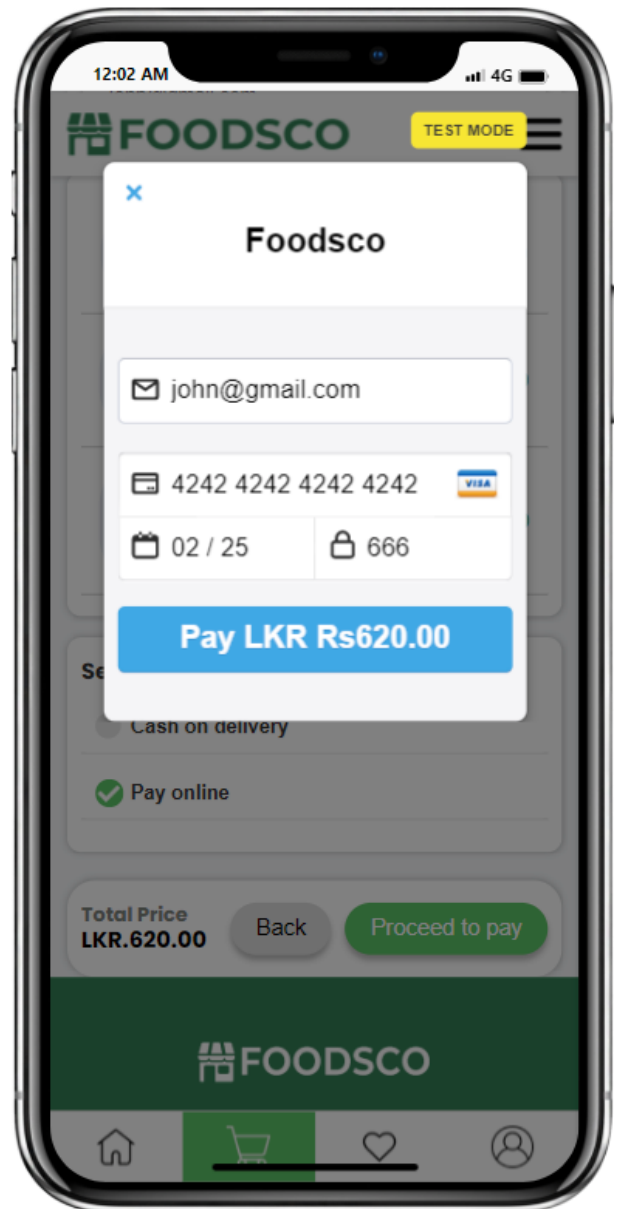
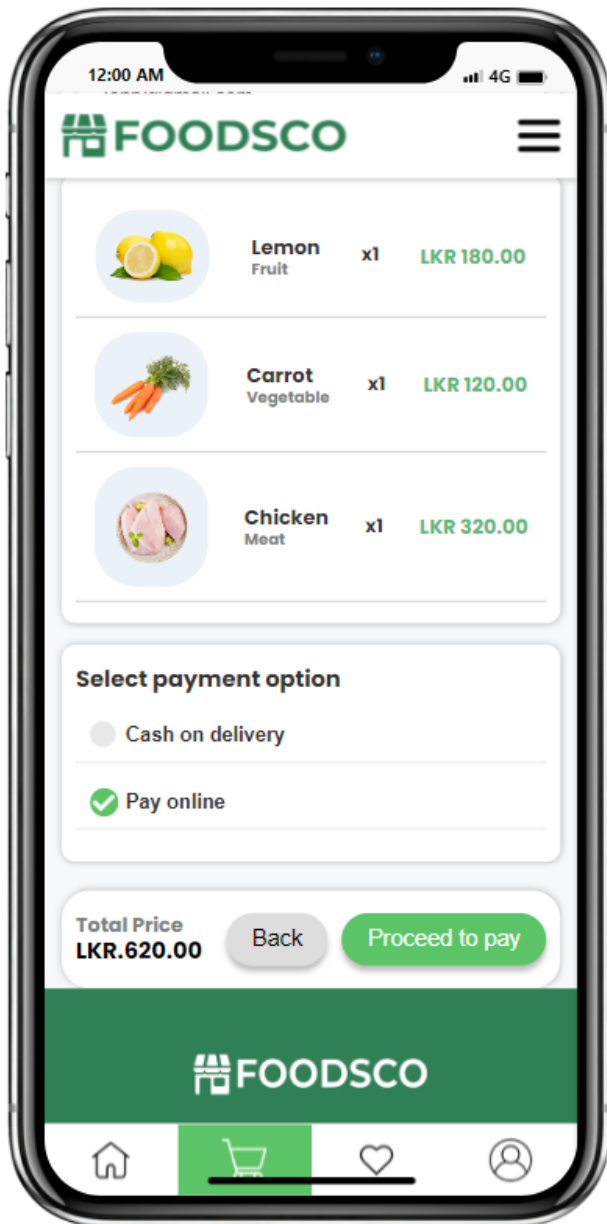
There is a difference in the appearance of the iPad compared to the iPhone.



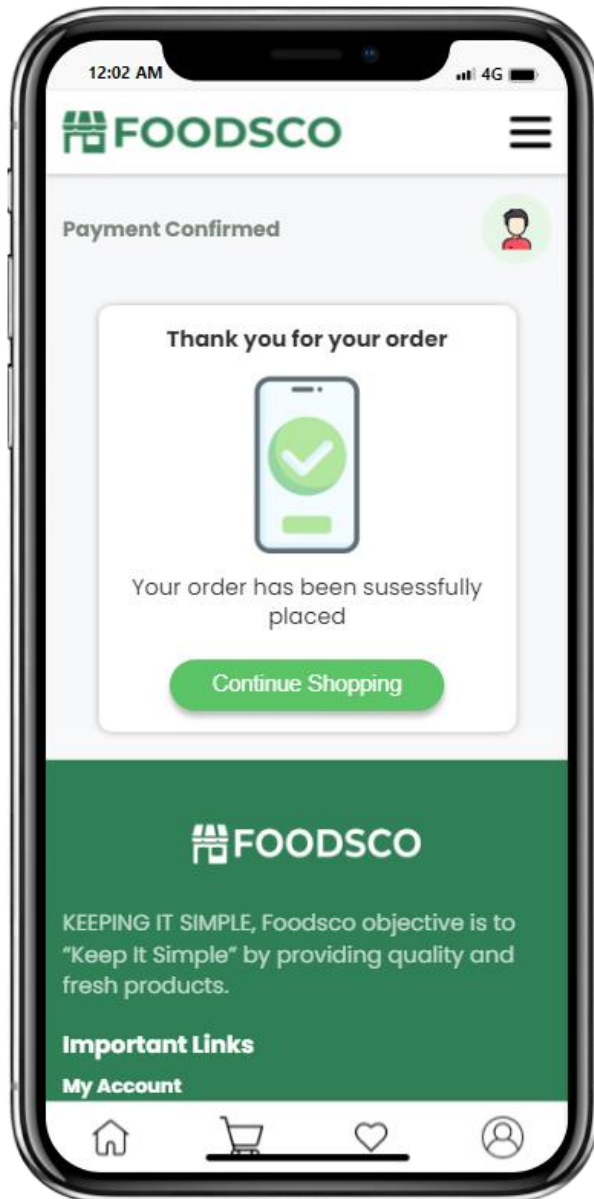
2.6 Booking form

How did you design and implement the booking form?





How do you design and implement booking form completion progress?



To implement the online payment method passing the correct calculation I used the total price of the products in the cart that were stored in the local storage and passed it to the online checkout form.


```

1  $(document).ready(function () {
2
3      let netTotal = localStorage.getItem("TotalPrice");
4      console.log(netTotal);
5
6      var path = window.location.pathname;
7      var page = path.split("/").pop();
8
9      jQuery(function($) {
10         var $form = $('#ctb-payment');
11         var handler = StripeCheckout.configure({
12             key: 'pk_test_cp21BcECf4kMMUbslRlZlsMo',
13             token: function(token) {
14
15                 if(token.id){
16                     if (page === "Checkout.html") {
17                         window.location.href = 'Payment-successful.html';
18                     }
19                 }
20             }
21         });
22
23
24         $('#place-order').on('click', function(e) {
25             // Code Section B  Open Checkout with further options
26             handler.open({
27                 name: 'Foodsco',
28                 currency: 'LKR',
29                 amount: netTotal * 100
30             });
31             e.preventDefault();
32         });
33
34         // Code Section C  Close Checkout on page navigation
35         $(window).on('popstate', function() {
36             handler.close();
37         });
38     });

```

Html

```
<div class="payment-option">
  <div class="po-heading">Select payment option</div>
  <div class="radio-wrapper">
    <label for="radio-choice-0" class="radio-label">Cash on delivery</label>
    <input type="radio" name="radio-choice-0" id="radio-choice-0" class="custom">
  </div>
  <div class="radio-wrapper-2">
    <label for="radio-choice-0b" class="radio-label">Pay online</label>
    <input type="radio" name="radio-choice-0" id="radio-choice-0b" class="custom">
  </div>
</div>

<div class="cart-total-box" id="ctb-payment">
  <div class="ctb-price-wrap">
    <div class="ctb-price-text">Total Price</div>
    <div class="ctb-price" id="single-price" >LKR 0.00</div>
  </div>

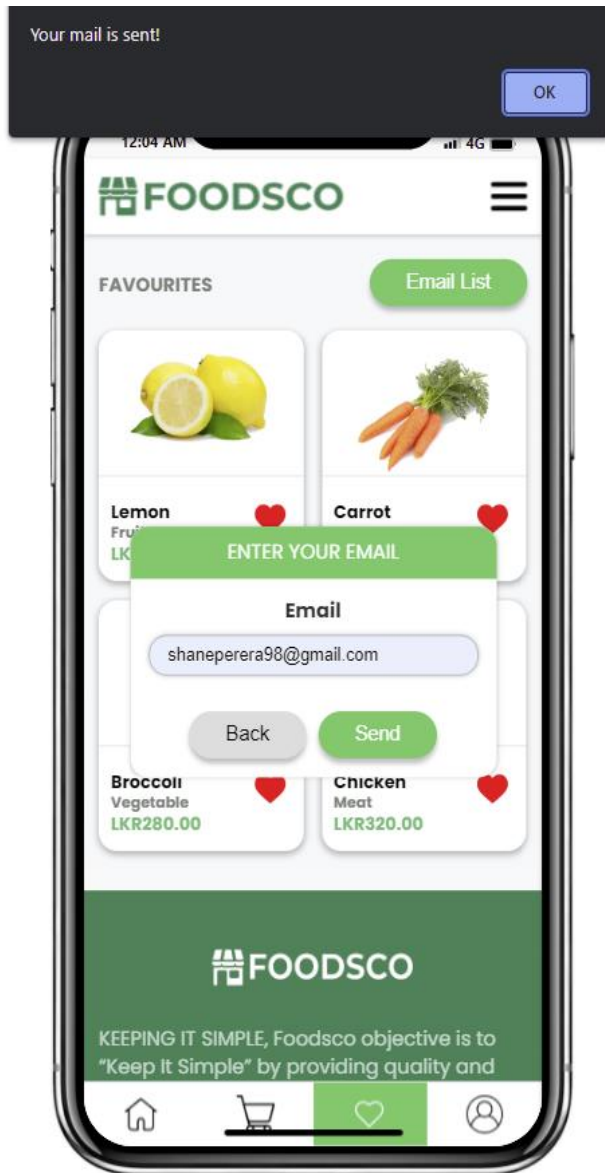
  <div class="btn-wrap">
    <div class="get-started-btn clear-btn">
      <button class="btn" ><a href="../pages/Cart.html" id="shopping-cart">Back</a></button>
    </div>
    <div class="get-started-btn">
      <button class="btn" id="place-order">Proceed to pay</button>
    </div>
  </div>
</div>
```

2.7 Favourites list

In the application the user can add products as favourites by clicking on the heart icon in the products page. After clicking the heart icon the id of the product is stored in the local storage. When the user navigates to the favorite page the products will be shown as below by retrieving from the local storage.

To send the favorite items list by email, we use EmailJS to send the email through client-side technologies. Because, EmailJS supports a lot of email services, mainly Gmail service.





```
// ***** Favourites *****

$(document).ready(function () {

    const IDs = localStorage.getItem("FavouritesList");
    var noOfItems;
    var FavouriteIDList;

    if (IDs.length != 0) {
        FavouriteIDList = IDs.split(",");
        noOfItems = FavouriteIDList.length;
    } else {
        FavouriteIDList = []
        noOfItems = IDs.length;
    }

    for (favouriteID of FavouriteIDList) {
        for (var i = 0; i < Products.length; i++) {
            if (Products[i].prod_id === favouriteID) {

                console.log(Products[i]);

                const CarHTML = '<li class="product">' +
                    '<a href="#">' +
                        '<div class="product-loop-custom-img-wrap">' +
                            '' +
                        '</div>' +
                        '<div class="product-content">' +
                            '<div hidden class="prod-id">' + Products[i].prod_id + '</div>' +
                            '<h2 class="product-title">' + Products[i].prod_name + '</h2>' +
                            '<h3 class="pro-sub-title">' + Products[i].prod_category + '</h3>' +
                            '<span class="price">LKR' + Products[i].prod_price + '</span>' +
                        '</div>' +
                    '</a>' +
                        '<div class="heart-wrapper">' +
                            '' +
                        '</div>' +
                    '</li>'

                document.getElementsByClassName('card-area')[0].innerHTML += CarHTML;
                break;
            }
        }
    }
})
```

```
$(".heart-icon").click(function () {  
  
    var prodID = $(' .prod-id').eq($('.heart-icon').index(this)).text()  
  
    for (var i = 0; i < FavouriteIDList.length; i++) {  
        if (FavouriteIDList[i] === prodID) {  
            FavouriteIDList.splice(i, 1);  
        }  
    }  
    localStorage.setItem('FavouritesList', FavouriteIDList);  
    location.reload();  
});
```

```

//***** favourite list email*****//

$('#enter-email').click(function () {
    $('#enterEmailPopup').addClass('open');
});

var emailProductNames = [];
var productName=$('#.product-title').text().split(/(?=[A-Z])/);

emailProductNames.push(productName);

localStorage.setItem('FavoriteProducts',emailProductNames);

var EmailItemList = localStorage.getItem("FavoriteProducts");

$('#submit-mail').click(function () {

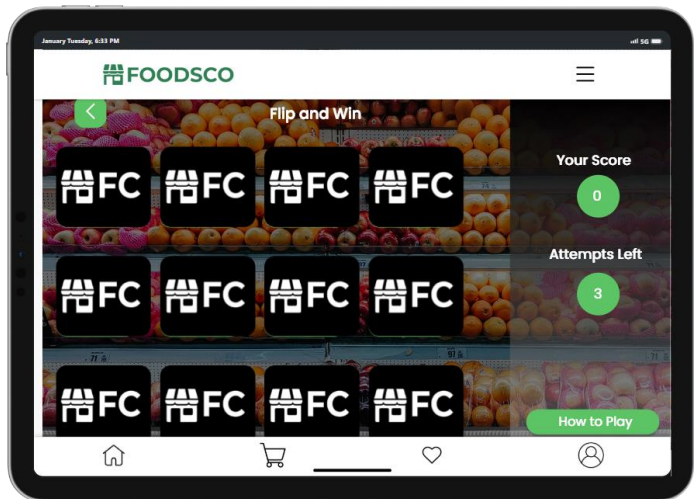
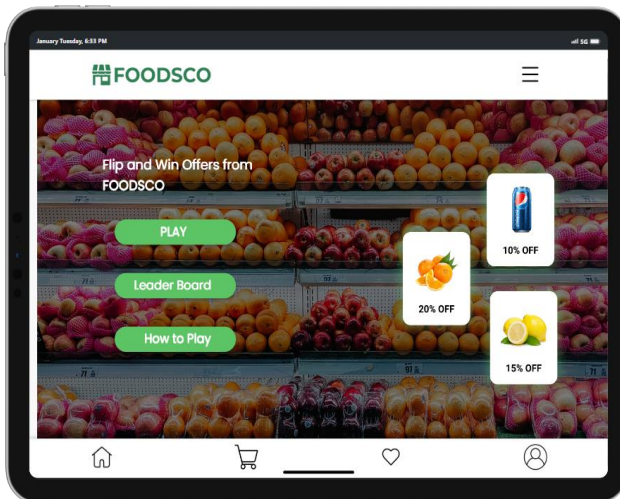
    var emailToSend = $('#mail-to').val();
    var data = {
        service_id: 'service_79j4rvn',
        template_id: 'template_vsbe24h',
        user_id: 'user_evdf04seM3aMERnwZnJZQ',
        template_params: {
            to_name: emailToSend,
            from_name: 'Foodsco',
            message_html: '<b>' + EmailItemList + '</b>'
        }
    };
    $.ajax('https://api.emailjs.com/api/v1.0/email/send', {
        type: 'POST',
        data: JSON.stringify(data),
        contentType: 'application/json'
    }).done(function() {
        alert('Your mail is sent!');
    }).fail(function(error) {
        alert('Oops... ' + JSON.stringify(error));
    });

});
});

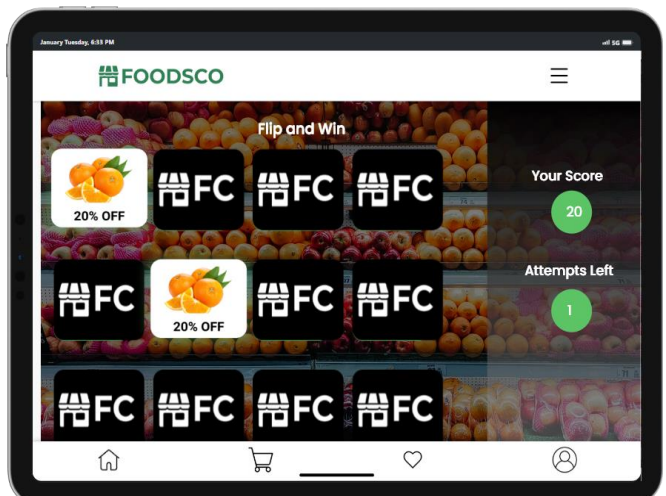
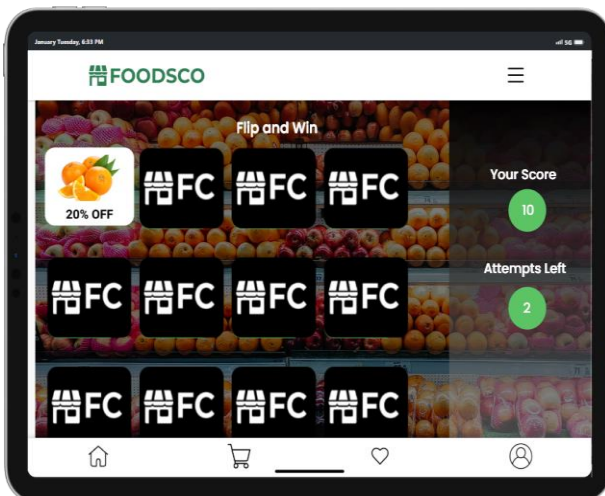
```

2.8 Gamification (might need to be combined with 2.9)

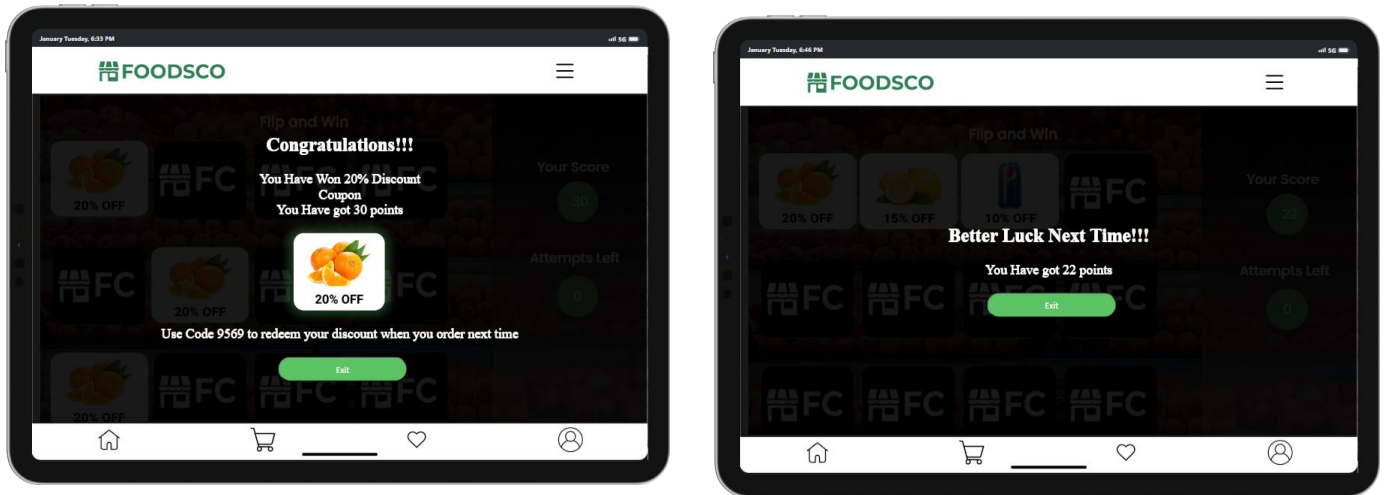
The gamification is a flip card game where the user gets 3 chances to flip the card and get discount coupons.



As the cards are flipped the score would increase and the number of attempts would decrease.

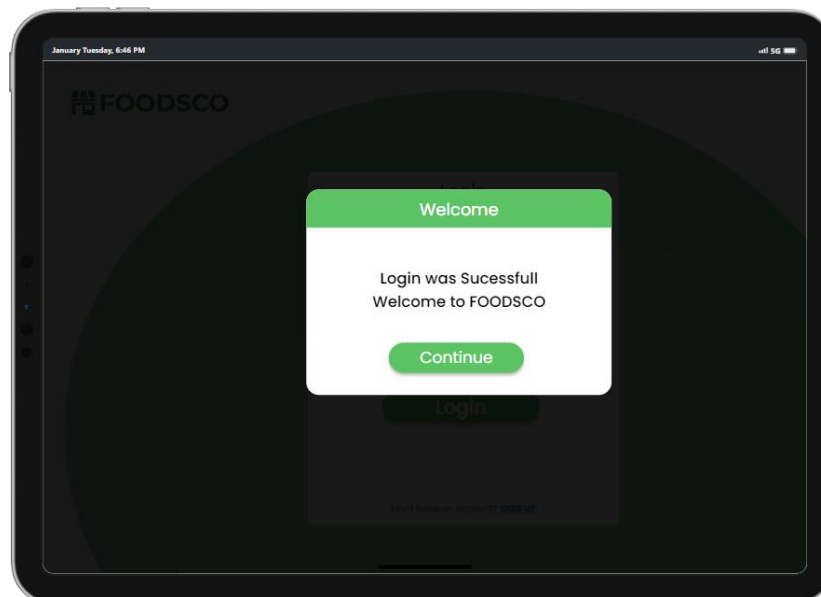


After all the attempts are completed, the results would be displayed.

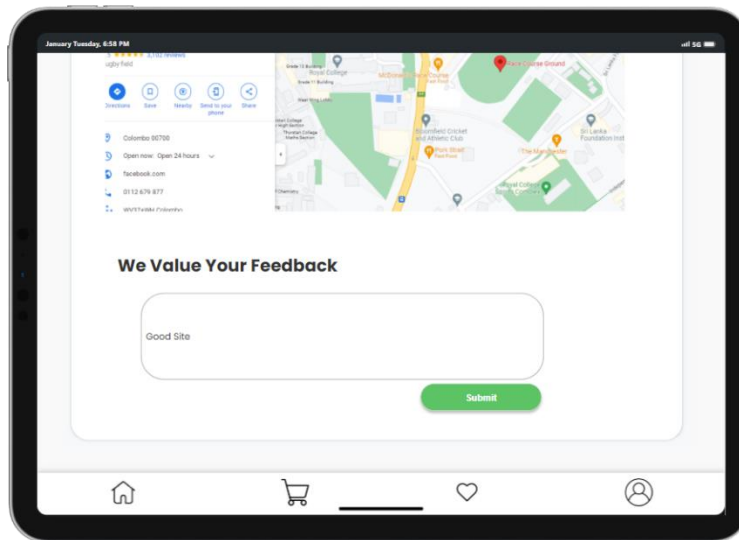


2.9 System feedback

System responses have been used in different places like in the Login and Sign Up.



The system is taking feedback from the user in the contact us page. The feedback is stored in an array that is stored in local storage.



```
function feedbackSubmit(){
  document.getElementById('submit-popup-overlay').style.display = 'block';
  document.getElementById('show-popup').style.display = 'block';

  console.log(document.getElementById("feedback").value);

  var feedback_msg =document.getElementById("feedback").value;

  feedbackMessages.push({"Name": "Thomas", "Feedback":feedback_msg});

  var popup = '<div class="contact-feedback-response">'+
    '<div class="popup-title"><p>Thank You</p></div>'+
    '<div class="popup-description"><p>Thank You for your<br> Feedback</p></div>'+
    '<div class="popup-btn">'+
    '  <button id="popup-continue" type="button" value="All" onClick="feedbackContinue()"><span>Continue</span></button>'+
    '</div>'+
    '</div>';

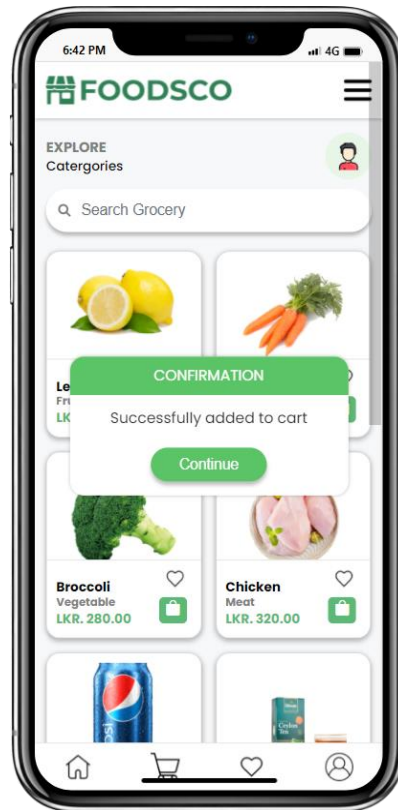
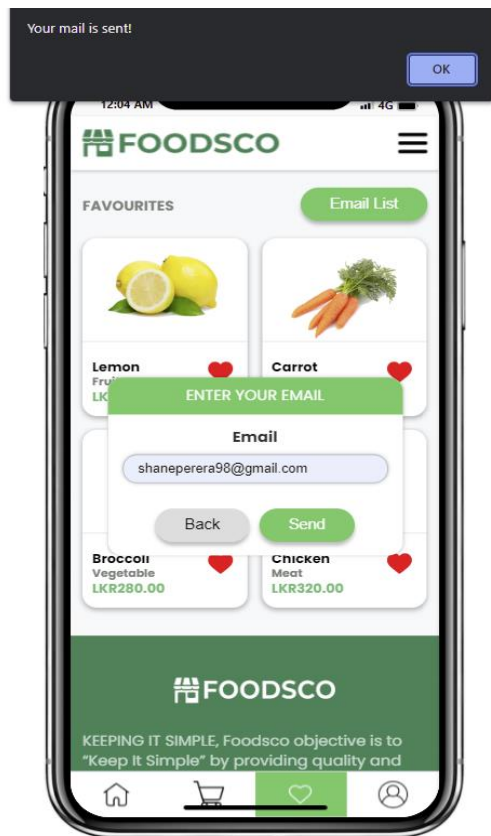
  $("#show-popup").html(popup);

  // Setting up the leader board to local storage
  localStorage.setItem('feedbackMessages',JSON.stringify(feedbackMessages));

  document.querySelector("#submit-popup-overlay").addEventListener("click", () => {
    document.getElementById('submit-popup-overlay').style.display = 'none';
    document.getElementById('show-popup').style.display = 'none';
    document.getElementById('feedback').value = '';
  });
}

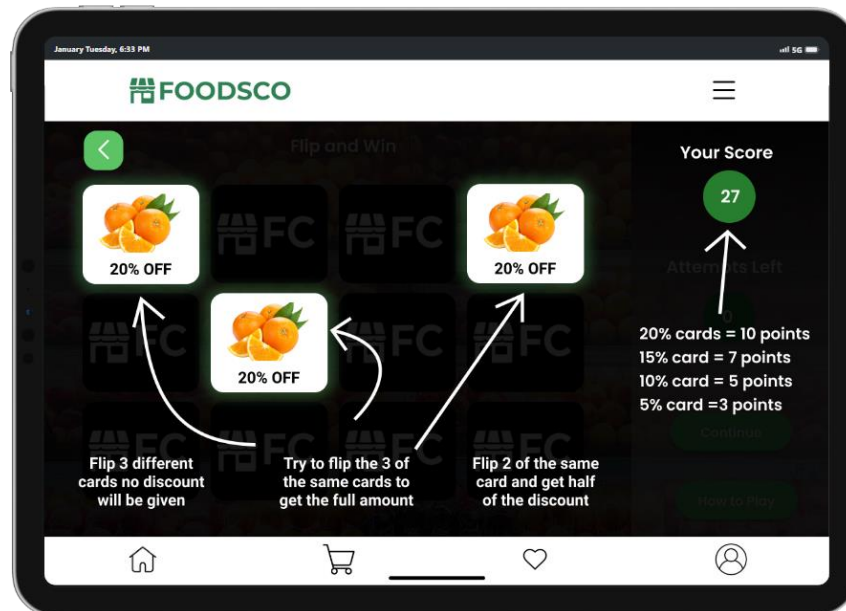
function feedbackContinue(){
  document.getElementById('submit-popup-overlay').style.display = 'none';
  document.getElementById('show-popup').style.display = 'none';
  document.getElementById('feedback').value = '';
}
```

Compared to iPad, in the iPhone the system responses alert boxes that will pop up after a certain task is completed or a message regarding the task.



2.10 Scores & points gained (might need to be combined with 2.7)

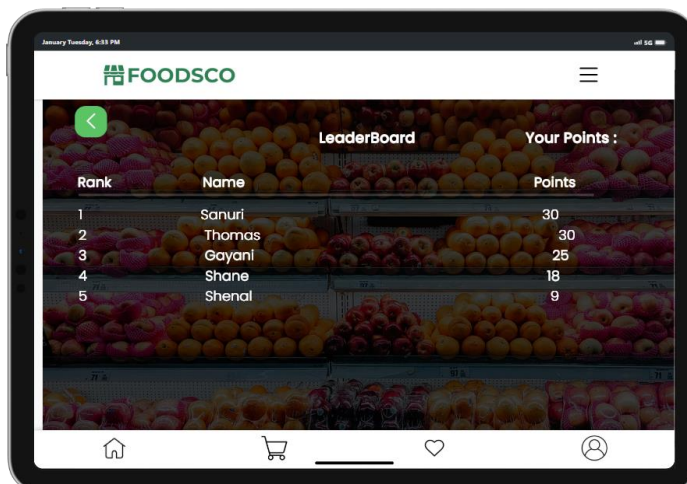
The points that are given in the game is displayed bellow and how to play the game to.



The point of each player is stored in a array which is stored in the local storage.

```
leaderBoard.push({"Name": name, "Score":total_points});
```

And the leaderboard would be displayed like shown below



2.11 Comments

In our application users are able to comment on the product by clicking the Tap to comment button. The username is captured as session storage in local storage from login page that helps the user to minimums the steps when writing the comment. The comment section includes the reply comment and additionally implemented like count to the relevant comment.

Customer Reviews

Add first review for this product

Tap to comments


Title:

Great Product

Review:

Fast Delivery. It's good. Keep it up 🍑

Cancel Submit

**Thomas@gmail.com**
Reviewed a product - 1/18/2022, 11:26:34 AM

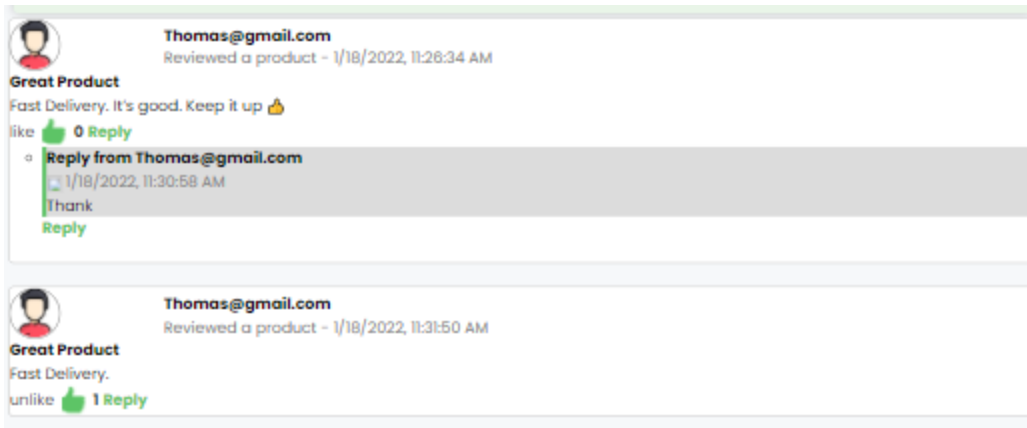
Great Product

Fast Delivery. It's good. Keep it up 🍑

like 🍑 0 Reply

Thank

Reply Cancel



HTML page

```
<div class="sp-sec">
  <div class="ui-content" style="padding: 5px;background: #E9F5E8;box-shadow: 0px 0px 10px rgb(0 0 0 / 25%);border-radius: 5px;">
    <span class="ui-grid-solo">
      <span class="ui-block-a">
        <span class="foodsco_title_" style="color: black ;font-weight: 700 ">Customer Reviews</span><br/><br/>
      </span>
    </span>
    <span class="ui-grid-a">
      <span class="ui-block-a" style="width: 35%" >
        <span class="foodsco_title_" style="color: black ;font-weight: 700 " id="reviewsCount"></span><br/><br/>
      </span>
      <span class="ui-block-b" style="width: 60%; ">
        <button id="show-form-btn"
          style="width:120px;float:right;color: #fff;border: 1px solid;
          font-size: 12px;font-weight: 400;margin: 5px;padding: 8px 7px 8px 7px;background-color: #5DC466;">
            Tap to comments</button>
        </span>
      </span>
    </span>
  </div>
</div>
```

```

<div class="ui-content" id="review-form" style="display: none; padding: 10px;background: #E9F5E8;
box-shadow: 0px 0px 10px rgb(0 0 0 / 25%);border-radius: 5px;margin-left: 15px;margin-right: 15px;">
  <div id="comment-form" >
    <span class="foodsco_subtitle_" style="color: black ; font-weight: 700;display: none" >Username:</span>
    <input type="text" name="name" id="name" value="" style="display: none">

    <span class="foodsco_subtitle_" style="color: black ; font-weight: 700;margin-left:5px">Title:</span>
    <input type="text" name="handle" id="handle"
      value="Review title" placeholder="Review title" style="background-color: white !important; ">

    <span class="foodsco_subtitle_" style="color: black ; font-weight: 700;margin-left:5px">Review:</span>
    <textarea rows="3" style="background-color: white !important;" id="comment" class="comment-box"
      placeholder="Body of review(1500)">Please write your comment</textarea>

    <div class="ui-grid-solo" >
      <span class="ui-block-a">
        <span style="display: flex;float:right" >
          <button id="hide-form-btn" class="cancel-btn" onclick="close_reply_form({abc})"
            style=" width:65px; padding: 8px 8px 8px 8px;color: #5DC466;background-color: white;
            border: 1px solid #5DC466;font-weight: 400;margin-right: 10px ;font-size: 12px;">Cancel</button>
          <button id="add-comment" class="add-btn" style="width:65px; color: white;
            background-color: #5DC466;border: 1px solid #5DC466;font-size: 12px;
            font-weight: 400;margin: 5px;padding: 8px 7px 8px 7px;">Submit</button>
        </span>
      </span>
    </span>
  </div>
</div>
<div class="ui-content" style="background-color: #F6F8FA;margin-top: 5px;">
  <ul id="commentslist" style="padding-inline-start:10px; padding-inline-end: 10px; background-color: #F6F8FA;"></ul>
</div>

```

JavaScript page

```

if(localStorage.getItem('products') == null){
  localStorage.setItem('products', JSON.stringify(productData))
}

let currentUser = JSON.parse(localStorage.getItem('users'));
console.log("App Started...");
console.log("Current User in Session: ",currentUser);

console.log("Check products : ", JSON.parse(localStorage.getItem('products')));

```

```
function create_comment(name,handle,comment,parent,email){
  console.log(comments_array.length,name,handle,comment,parent,email);
  var comment = string_to_json(comments_array.length,name,handle,comment,parent,email)

  comments_array.push(comment)
  if(parent != null) {
    comments_array[parent].childrenIds.push(comments_array.length-1);
  }
  store_comments()
  show_comments()
}

//create comment object
function string_to_json(id,name,handle,comment,parent,email){
  var commentObj = {}
  // var today = new Date();
  commentObj['id'] = id
  commentObj['name'] = name
  commentObj['email'] = email
  commentObj['handle'] = handle
  commentObj['comment'] = comment
  commentObj['parentId'] = parent
  // commentObj['lastUpdated'] = String(new Date()).split("T")[0]
  commentObj['lastUpdated'] = new Date().toLocaleString();
  commentObj['childrenIds'] = []

  var json = (commentObj);

  return json
}
```

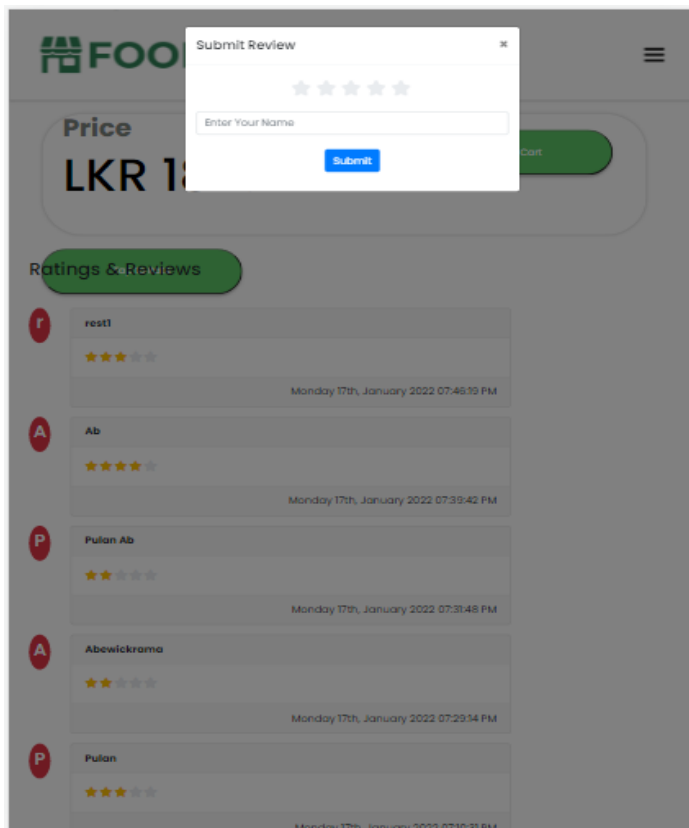
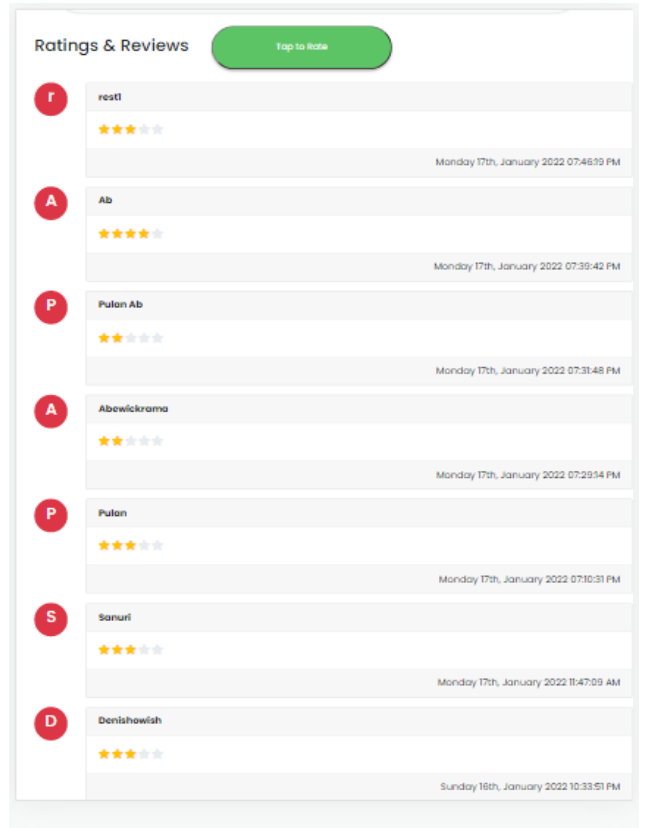
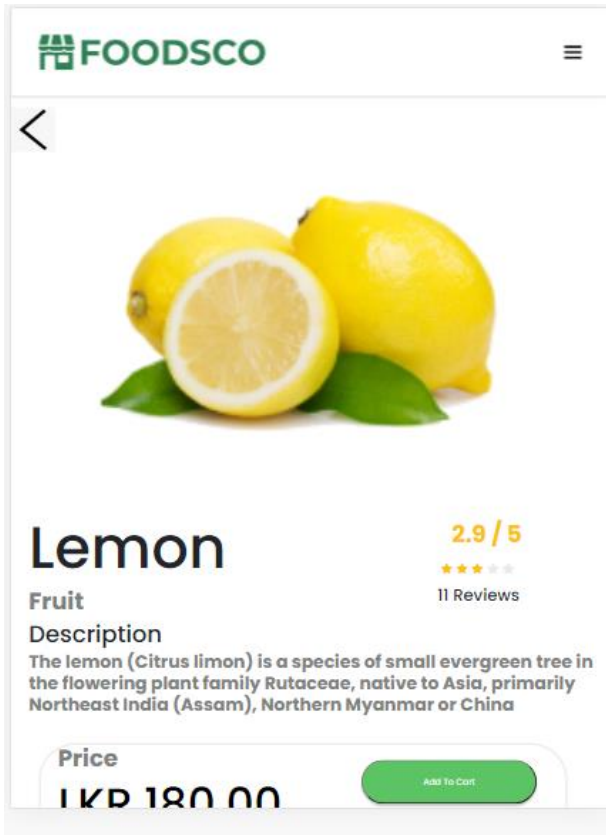


```
function store_comments() {  
  
    var temp = new Array()  
    for (var i = 0; i < comments_array.length; i++) {  
        temp.push(comments_array[i])  
    }  
  
    // var urlString = window.location.href;  
    // var url = new URL(urlString);  
  
    var product_id = "1";  
    console.log("id = " + product_id);  
  
    var product_stored_obj = localStorage.getItem('products');  
    var products = JSON.parse(product_stored_obj)  
    console.log('products',products)  
    //check the prod_id  
    for ( let i =0 ; i < products.length ; i ++ ) {  
        console.log('--',products[i]['id'],product_id)  
  
        if(products[i]['prod_id'] == product_id ) {  
            // console.log('found ... ')  
            console.log(products[i]['comments'])  
            products[i]['comments'] = temp  
            console.log(products[i]['comments'])  
            // console.log('found end ... ')  
        }  
    }  
  
    console.log(products)  
    localStorage.setItem('products', JSON.stringify(products));  
}
```

```

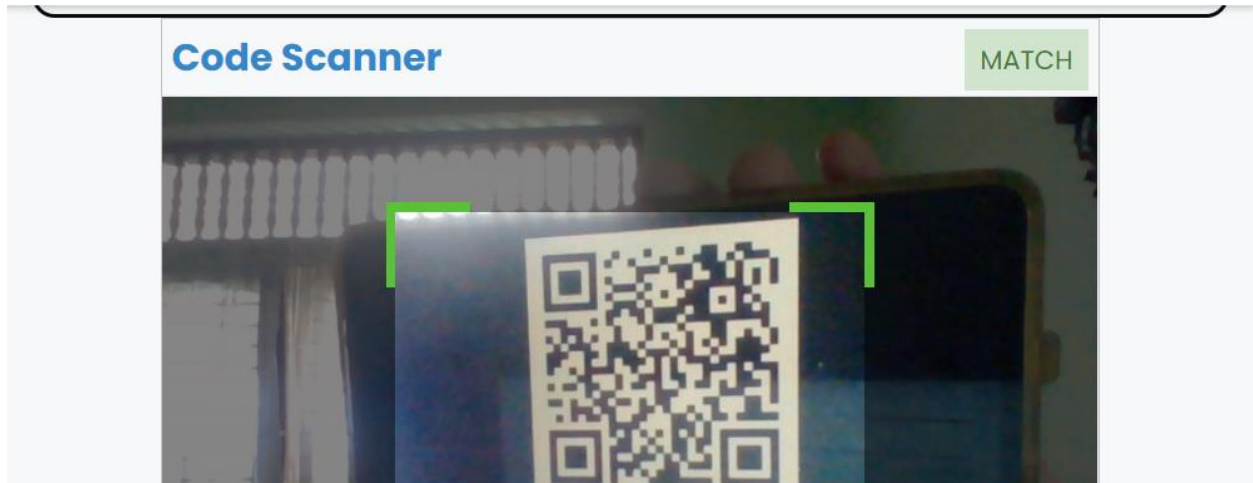
function show_comment(comment){
  // console.log(comment)
  if(comment.parentId == null) {
    // console.log('here 1 ')
    //parent comment
    var display_elem =
    `
    <div class="ui-contents" style="border: 1px solid #C4C4C4; background-color:white; border-radius: 5px ;">
      <li id="comment-${comment.id}">
        <div class="ui-grid-a">
          <!-- user data -->
          <div class="ui-block-a" style="width: 25%">
            
          </div>
          <div class="ui-block-b"
            style="
              width: 75%;
              padding: 0;
              margin-top: 5px;
              margin-left: -100px;
              margin-bottom: 0;
              margin-right: 0;">
            <span class="foodsko_title" style="color: black;font-weight: 700">${comment.name}</span><br/>
            <span class="foodsko_content" style="color: grey !important"> Reviewed a product - ${comment.lastUpdated}</span>
          </div>
          <span class="foodsko_title" style="color: black;font-weight: 700" >${comment.handle}</span><br/>
          <span class="foodsko_content" style="color: black">${comment.comment}</span><br/>
          <span class="btn-like" id="like-${comment.id}">${comment.like == true ? 'unlike': 'like'}</span>
        `
  }
}

```



2.13 AR/QR

We implemented a QR scanner to direct users to the poi page. When the user scans the QR code the relevant URL is shown as a result button, so the user is able to see the URL to share among the friends. Also, user can direct to the relevant page by clicking on the result button.



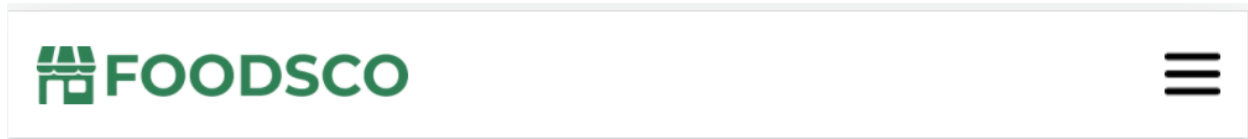
```
<script type="text/javascript">
  function onScanSuccess(qrCodeMessage) {

    document.getElementById('result').innerHTML = '<span class="result" style="" id="urlBtn" onclick="myFunction()">'
      +qrCodeMessage+'</span>';

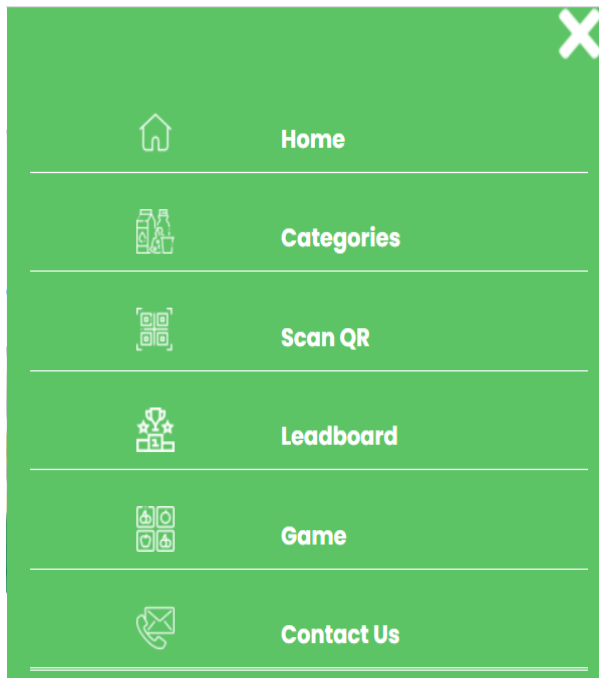
    // location.replace(qrCodeMessage);
  }
  function myFunction() {
    var content = document.getElementById('urlBtn').innerHTML;
    console.log("QR button");
    location.replace(content);
  }
  > function onScanError(errorMessage) { ...
  var html5QrcodeScanner = new Html5QrcodeScanner("reader", { fps: 10, qrbox: 250 });
  html5QrcodeScanner.render(onScanSuccess, onScanError);
</script>
```

2.14 Navigation

Also we developed a single navigation bar in all of our sites for app navigation so that the user can simply move to different areas of the applications. With the use of the hamburger symbol, it is simple to move between the pages of the apps.



```
<div data-role="header">
  <nav class="navbar navbar-fixed-top navbar-default" id="site-header">
    <div class="container">
      <div class="flex-parent">
        <div class="navbar-header">
          <div class="navbar-brand-wrp">
            <a class="navbar-brand header-logo" href="#" rel="external">
              
            </a>
          </div>
          <div class="ham-icon-wrp">
            <a class="ham-icon" href="#" rel="external">
              
            </a>
          </div>
        </div>
      </div>
    </div>
  </nav>
```



```
<div class="menu-btn">
  <a href="./Home.html" rel="external">
    <div class="ui-block-a menu-icon-position">
      
    </div>
    <div class="ui-block-b menu-font-position">
      <span class="menu-font">Home</span>
    </div>
  </a>
</div>
```

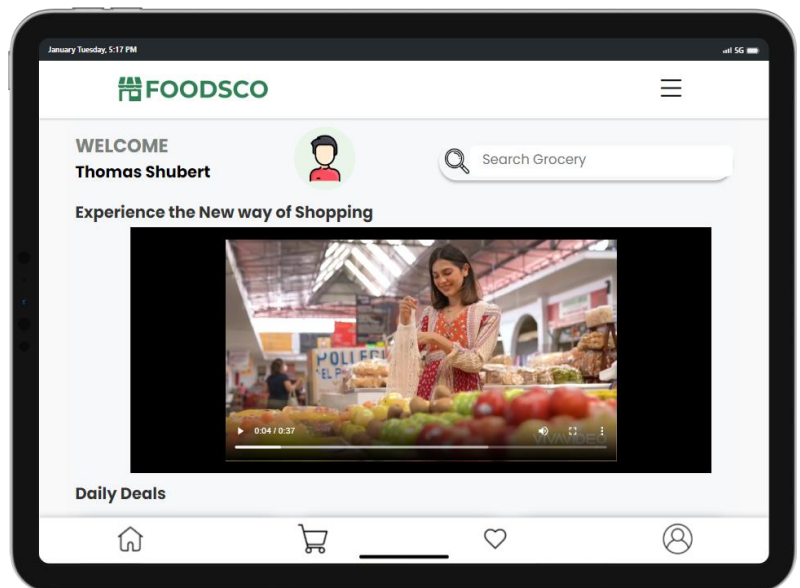
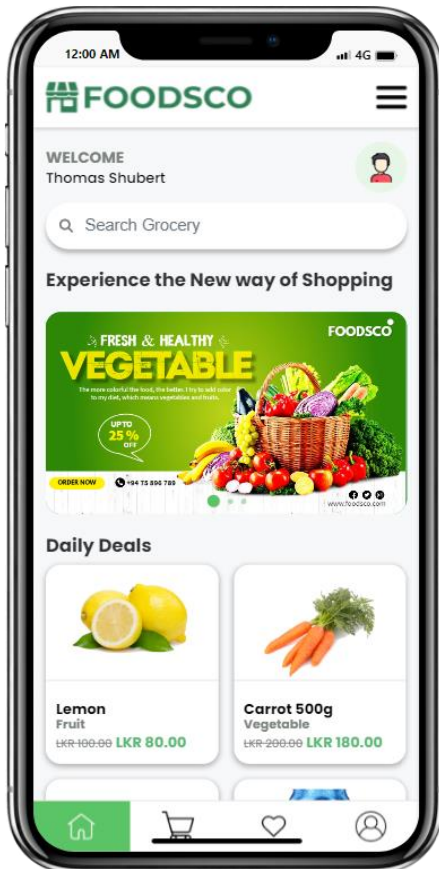


```
<!-- bottom fixed navigation -->
<div class="footer-nav">
  <div class="fn-wrap">
    <div class="fn-item current">
      <a href="./Home.html" rel="external">
        
      </a>
    </div>
    <div class="fn-item">
      <a href="../pages/Cart.html" rel="external">
        
      </a>
    </div>
    <div class="fn-item">
      <a href="../pages/Favourites.html" rel="external">
        
      </a>
    </div>
    <div class="fn-item">
      <a href="#">
        
      </a>
    </div>
  </div>
</div>
```

A navigation bar was added in the bottom as it follows the Tabbed view pattern. The tabbed views give a smooth, self-evident, and difficult to miss nature while letting the user reach the primary components via the user's thumb zone.

2.15 Differentiation between iPhone and iPad implementation

For the introductory section we design and implemented an image slider for the iPhone and for the iPad we inserted a video for the same section



References

Icons - <https://www.flaticon.com/free-icons/online-store>

Images - <https://www.cleanpng.com/>

Images - <https://www.pexels.com/>

Colors - <https://coolors.co/>

[Designing The Information Architecture \(IA\) of Mobile Apps | Applikey \(applikeysolutions.com\)](#)

<https://www.emailjs.com/>

<https://reeteshghimire.com.np/2021/05/27/access-camera-and-read-qr-code-using-javascript/>

<https://demos.jquerymobile.com/1.4.5/>