

## STAR RATING



### What is our GOAL for this MODULE?

We learned to add star rating functionality in the AR Menu Card scene and to update the dish rating and review in the firestore database.

### What did we ACHIEVE in the class TODAY?

- Learned to use Bootstrap to add star rating and review for a particular dish.
- Learned to update the database to add the dish rating and review.

### Which CONCEPTS/CODING BLOCKS did we cover today?

- Firebase as database
- swal()
- Bootstrap rating libraries, <textarea>, <input>
- HTML Modal
- ngrok to run the application.

### How did we DO the activities?

1. Add the Bootstrap libraries to index.html.

```
<link href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.0/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css" />
<script src="//maxcdn.bootstrapcdn.com/bootstrap/3.3.0/js/bootstrap.min.js"></script>
<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>
```

2. Add a Modal Box to show the pop-up having a text field to write a review, add star rating inputs and a submit button to save them in the database on click.
  - Update **index.html**.

```
<!-- Boilerplate for Ratings Modal -->
<div class="container">
  <div id="rating-modal-div" class="rating-modal" tabindex="-1" role="dialog">
    <div class="modal-dialog" role="document">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-title">Rate Dish</h5>
          <button type="button" class="close" data-dismiss="modal" aria-label="Close" onclick="closeRatingModal()">
            <span aria-hidden="true">&times;</span>
          </button>
        </div>
        <div class="modal-body">
          <div class="table-responsive">
            <!--Text Field Input-->
            <textarea class="form-control" id="feedback-input" rows="3"></textarea>

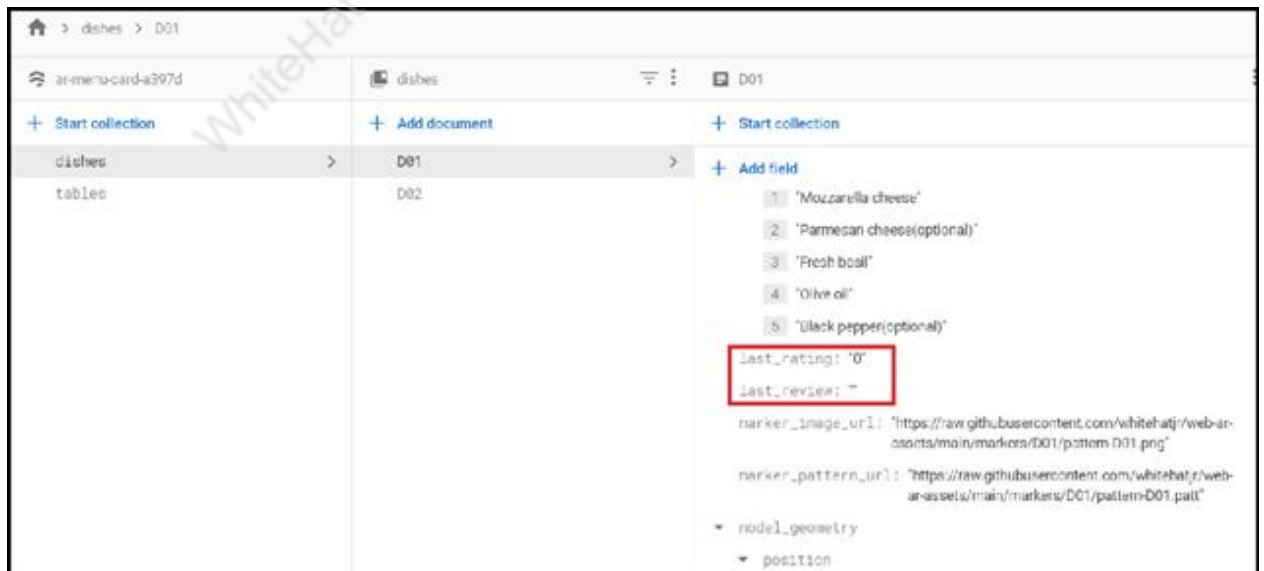
            <!--Stars Input-->
            <input id="rating-input" name="rating-input" class="rating rating-loading" data-min="0" data-max="5"
              data-step="0.1" value="0" />
          </div>
        </div>
        <div class="modal-footer">
          <!--Submit Button-->
          <button id="save-rating-button" type="button" class="btn btn-primary">
            Submit
          </button>
        </div>
      </div>
    </div>
  </div>
</div>
```

- style.css (initial display property is set none).

```
.rating-modal {
  display: none; /* Hidden by default */
  position: fixed; /* Stay in place */
  z-index: 1; /* Sit on top */
  padding-top: 100px; /* Location of the box */
  left: 0;
  top: 0;
  width: 100%; /* Full width */
  height: 100%; /* Full height */
  overflow: auto; /* Enable scroll if needed */
  background-color: #rgb(0,0,0); /* Fallback color */
  background-color: #rgba(0,0,0,0.4); /* Black w/ opacity */
}

/*!
Styling for Stars
*/
.rating-loading{width:25px;height:25px;font-size:0;color:#fff;background:url(../img/loading.gif) t
```

3. Create a JavaScript file(star.js) and add JS Code from [here](#).
4. Add the required fields to store rating and review in the database.



5. For the “RATE DISH” button, update the click event listener to call the **handleRatings()** function.

```
// Handling Click Events
ratingButton.addEventListener("click", () => this.handleRatings(dish));

orderButton.addEventListener("click", () => {
  var tNumber;
  tableNumber <= 9 ? (tNumber = `T0${tableNumber}`) : `T${tableNumber}`;
  this.handleOrder(tNumber, dish);

  swal({
    icon: "https://i.imgur.com/4NZ6uLY.jpg",
    title: "Thanks For Order !",
    text: "Your order will serve soon on your table!",
    timer: 2000,
    buttons: false
  });
});

orderSummaryButton.addEventListener("click", () =>
  this.handleOrderSummary()
);

payButton.addEventListener("click", () => this.handlePayment());
}
},
handleRatings: async function(dish) {
},
```

6. Query the database to check if the dish is present in **current\_orders** in the database.

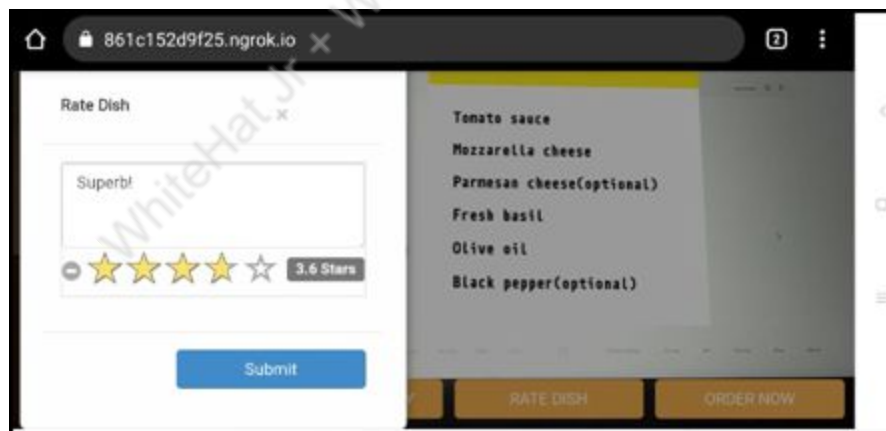
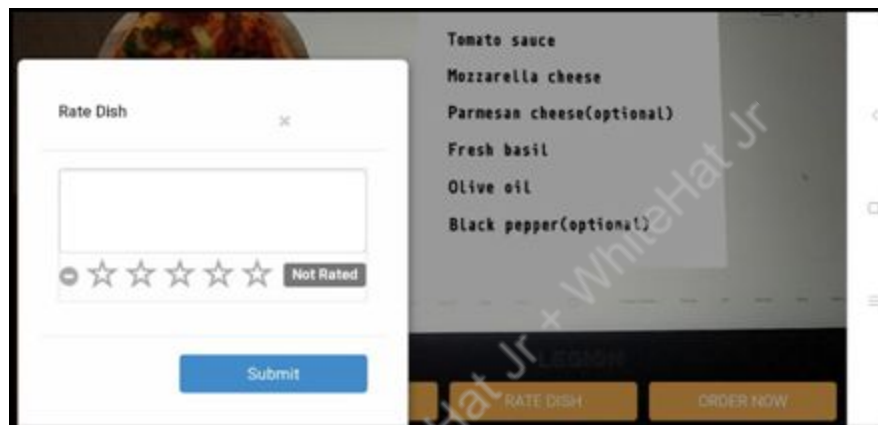
```
handleRatings: async function (dish) {  
  
    // Getting Table Number  
    var tNumber;  
    tableNumber <= 9 ? (tNumber = `T0${tableNumber}`) : `T${tableNumber}`;  
  
    // Getting Order Summary from database  
    var orderSummary = await this.getOrderSummary(tNumber);  
  
    var currentOrders = Object.keys(orderSummary.current_orders);  
  
    if (currentOrders.length > 0 && currentOrders==dish.id) {  
  
        //Submit button click event  
  
        //Get the input value(Review & Rating)  
  
        //Update db  
  
    }  
    else{  
        //Show alert  
  
    }  
  
},
```

7. Save the rating input and feedback input in the database and show an alert after it is saved.

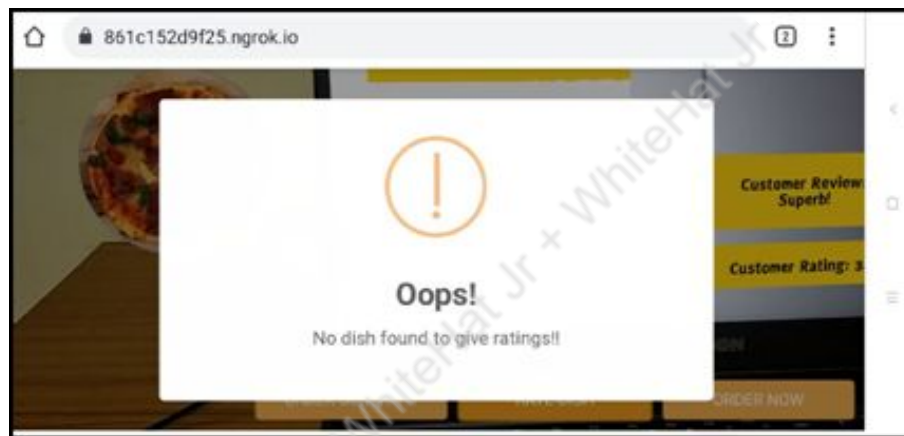
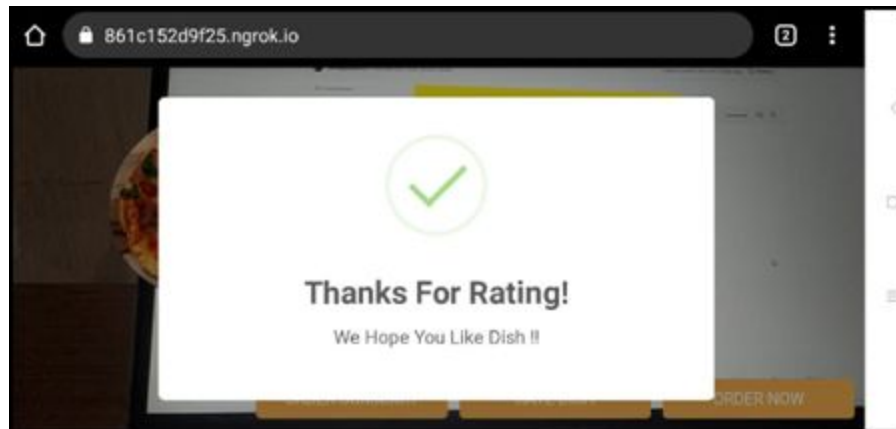
```
if (currentOrders.length > 0 && currentOrders==dish.id) {  
  
  // Close Modal  
  document.getElementById("rating-modal-div").style.display = "flex";  
  document.getElementById("rating-input").value = "0";  
  document.getElementById("feedback-input").value = "";  
  
  //Submit button click event  
  var saveRatingButton = document.getElementById("save-rating-button");  
  
  saveRatingButton.addEventListener("click", () => {  
    document.getElementById("rating-modal-div").style.display = "none";  
    //Get the input value(Review & Rating)  
    var rating = document.getElementById("rating-input").value;  
    var feedback = document.getElementById("feedback-input").value;  
  
    //Update db  
    firebase  
      .firestore()  
      .collection("dishes")  
      .doc(dish.id)  
      .update({  
        last_review: feedback,  
        last_rating: rating  
      })  
      .then(() => {  
        swal({  
          icon: "success",  
          title: "Thanks For Rating!",  
          text: "We Hope You Like Dish !!",  
          timer: 2500,  
          buttons: false  
        });  
      });  
  });  
} else{  
  swal({  
    icon: "warning",  
    title: "Oops!",  
    text: "No dish found to give ratings!!",  
    timer: 2500,  
    buttons: false  
  });  
}
```

## 8. Test the output using ngrok:

- Scan the dish marker.
- Click on “RATE DISH”.
- Write a review.
- Rate.
- Click on submit.
- Verify the update in the database.
- Test the output for the dish which is not added in the current\_order.









9. Add the plane entity to show the rating and review as text in the AR scene.

```
// Dish review plane
var reviewPlane = document.createElement("a-entity");
reviewPlane.setAttribute("id", `review-plane-${dish.id}`);
reviewPlane.setAttribute("position", { x: 2, y: 0, z: 0 });
reviewPlane.setAttribute("geometry", {
  primitive: "plane",
  width: 1.5,
  height: 0.5
});

reviewPlane.setAttribute("material", {
  color: "#F0C30F"
});
reviewPlane.setAttribute("rotation", { x: -90, y: 0, z: 0 });
reviewPlane.setAttribute("visible", false);

// Dish review
var review = document.createElement("a-entity");
review.setAttribute("id", `review-${dish.id}`);
review.setAttribute("position", { x: 0, y: 0.05, z: 0.1 });
review.setAttribute("rotation", { x: 0, y: 0, z: 0 });
review.setAttribute("text", {
  font: "monospace",
  color: "black",
  width: 2.4,
  align: "center",
  value: `Customer Review: \n${dish.last_review}`
});

reviewPlane.appendChild(review);
marker.appendChild(reviewPlane);
```

```
// Dish Rating plane
var ratingPlane = document.createElement("a-entity");
ratingPlane.setAttribute("id", `rating-plane-${dish.id}`);
ratingPlane.setAttribute("position", { x: 2, y: 0, z: 0.5 });
ratingPlane.setAttribute("geometry", {
  primitive: "plane",
  width: 1.5,
  height: 0.3
});

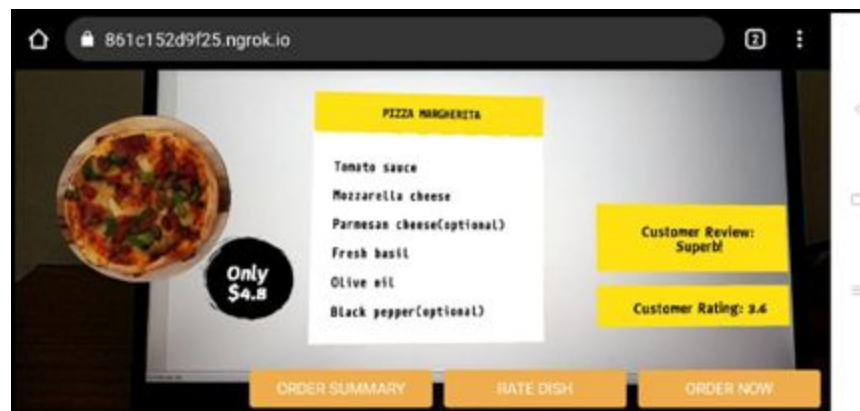
ratingPlane.setAttribute("material", {
  color: "#F0C30F"
});

ratingPlane.setAttribute("rotation", { x: -90, y: 0, z: 0 });
ratingPlane.setAttribute("visible", false);

// Ratings
var rating = document.createElement("a-entity");
rating.setAttribute("id", `rating-${dish.id}`);
rating.setAttribute("position", { x: 0, y: 0.05, z: 0.1 });
rating.setAttribute("rotation", { x: 0, y: 0, z: 0 });
rating.setAttribute("text", {
  font: "mozillavr",
  color: "black",
  width: 2.4,
  align: "center",
  value: `Customer Rating: ${dish.last_rating}`
});

ratingPlane.appendChild(rating);
marker.appendChild(ratingPlane);
```

10. Test the output using ngrok:



We learned how to use Bootstrap to add a star rating for a dish in augmented reality and to store the values into the database.

### What's NEXT?

In the next class, we will learn about barcode markers in augmented reality.

### EXTEND YOUR KNOWLEDGE:

- You can refer to the link below to explore more about AFrame  
[A-Frame](#)
- You can refer to the link below to explore more about HTML modal  
[HTML modal](#)
- You can refer to the link below for more examples of Bootstrap Rating  
[Bootstrap Rating](#)