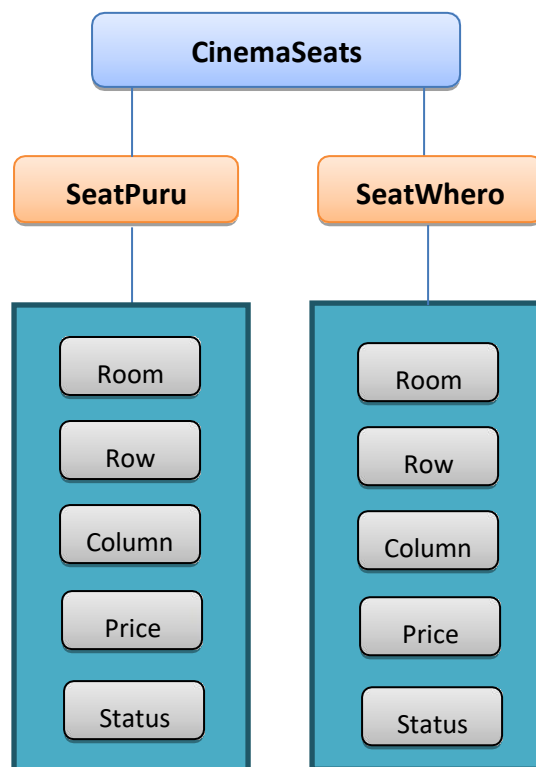


Project Documentation

Cinema Room

- **XML data design**

<CinemaSeats> is the root element of the xml and it is the parent tag of <SeatPuru> and <SeatWhero> elements which represent two different cinema rooms. Each tag has five child elements including room, row, column, price, and status.



1. Room

Storing the name of the room Puru or Whero in this tag.

<Room>Puru</Room>

<Room>Whero</Room>

2. Row

A, B, C and so on are the possible values of the element.

`<Row>A</Row>`

3. Column

Number 0 represents the seat is empty.

`<Column>0</Column>`

Numbers from 1 to 10 represent the sequence of the columns.

`<Column>1</Column>`

4. Price

There are four different prices.

0 means there is no seat.

`<Price>0</Price>`

20 the first two rows A and B.

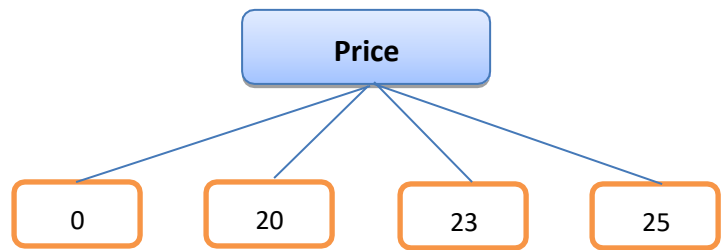
`<Price>20</Price>`

23 the rows from C to E.

`<Price>23</Price>`

25 the rest of the rows.

`<Price>25</Price>`



5. Status

There are three different status.

0 the seat is available

`<SeatPuru>`

`<Room>Puru</Room>`

`<Row>G</Row>`

`<Column>9</Column>`

`<Price>25</Price>`

`<Status>0</Status>`

`</SeatPuru>`

1 the seat is unavailable

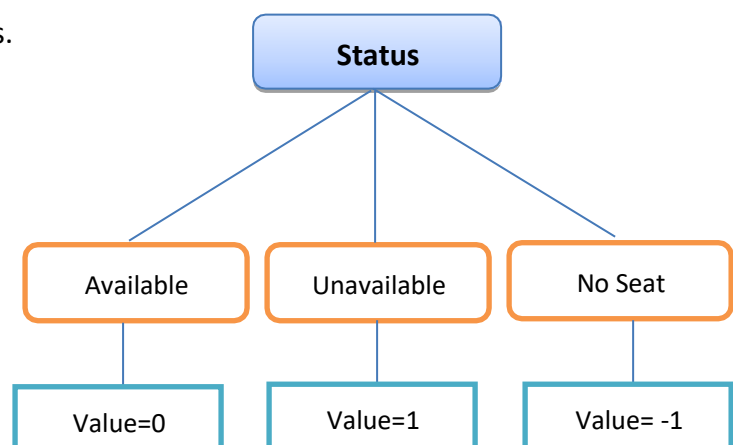
`<SeatWhero>`

`<Room>Whero</Room>`

`<Row>B</Row>`

`<Column>1</Column>`

`<Price>20</Price>`



```

        <Status>1</Status>
    </SeatWhereo>
    -1no seat
    <SeatPuru>
        <Room>Puru</Room>
        <Row>H</Row>
        <Column>0</Column>
        <Price>0</Price>
        <Status>-1</Status>
    </SeatPuru>

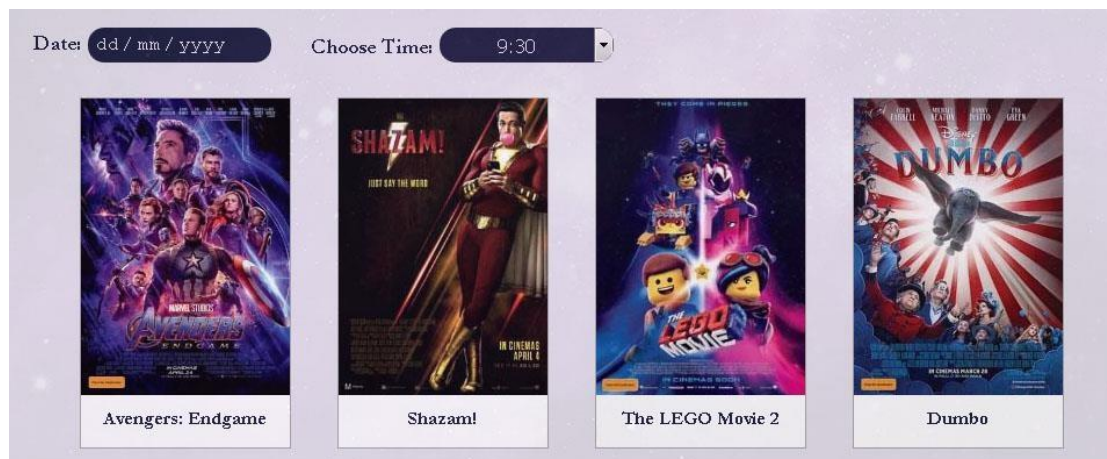
```

- **User booking design page**

1. Select Date, time and movie

Step 1: Select a date from calendar and select a time from dropdown box.

Step 2: Select a movie-----jump to room layout web.



Date, time and movie name are passing to room layout web by URL parameters.

Set URL parameters

```

function currentSelect(ele){
    if(currentDate.value!=''){
        ele.children[0].href="seatsRoom.html?movie="+ele.title+"&date="+
        currentDate.value+"&time="+currentTime.value;
    }else{
        alert('Please choose a date!');
        ele.children[0].href="";
    }
}

```

Get URL parameters

```
function getParam(){
    var url=window.location.href;
    var result = url.split("?")[1];
    var keyValue = result.split("&");
    var results=[];
    for (var i = 0; i < keyValue.length; i++) {
        var item = keyValue[i].split("=");
        results[i]=item[1];
    }
    results[0]=results[0].replace(/%20/g," ");
    movieName=results[0];
    dateTime=" "+results[1]+" "+results[2];
    disMovie.innerHTML=" "+results[0];
    disDate.innerHTML=" "+results[1]+" "+results[2];
}
```

2. Create dynamic table

Step 1: Create a seat object

```
function seat(room, row, column, price, condition){
    this.room=room;
    this.row=row;
    this.column=column;
    this.price=price;
    this.condition=condition;
}
```

Step 2: Access the XML document

```

var seatsPuru=[];
var seatsWhere=[];
if(window.XMLHttpRequest){
    xmlhttp=new XMLHttpRequest();
}
else{
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP")
}
xmlhttp.open("GET","seats.xml",false);
xmlhttp.send();
xmlDoc=xmlhttp.responseXML;

var x = xmlDoc.getElementsByTagName("SeatPuru");
var y = xmlDoc.getElementsByTagName("SeatWhere");
var room='';
var row='';
var column='';
var price=0;
var condition='';

```

Step 3: Load seats from xml and store in array

```

function loadSeats(currentTag, arraySeats){//get seats from xml
    var cols=1;
    var rows=0;
    for(i=0;i<currentTag.length;i++){
        room = currentTag[i].getElementsByTagName("Room")[0].childNodes[0].nodeValue;
        row=currentTag[i].getElementsByTagName("Row")[0].childNodes[0].nodeValue;
        column=currentTag[i].getElementsByTagName("Column")[0].childNodes[0].nodeValue;
        price=currentTag[i].getElementsByTagName("Price")[0].childNodes[0].nodeValue;
        condition=currentTag[i].getElementsByTagName("Status")[0].childNodes[0].nodeValue;
        var currentSeat=new seat(room, row, column, price, condition);
        arraySeats.push(currentSeat);
        if(row=='A'){
            cols++;
        }
        if(i==currentTag.length-1){
            rows=getRows(row);
        }
    }
    var results=new Array(rows,cols);
    return results;
}

```

Step 4: Dynamic create seats

```

function creatRoom(tableName,seats,idName,rows,cols){//creat table
    var table = document.createElement("table");
    table.setAttribute("id", tableName);
    if(tableName=="tableWhereo"){
        table.setAttribute("style", "display:none");
    }
    var flag=0;

    for(i=0;i<rows;i++){
        var currentRow=table.insertRow(i);
        var currentCells=[];

        for(j=0;j<cols;j++){

            currentCells.push(currentRow.insertCell(j));
            printRow(i,currentCells);
            if(flag<seats.length&&j!=0){
                seatClass(currentCells,j,seats[flag].condition);
                currentCells[j].setAttribute('seatNumber', seats[flag].row + seats[flag].column);
                currentCells[j].setAttribute('seatStatus', seats[flag].condition);
                currentCells[j].setAttribute('seatPrice', seats[flag].price);
                currentCells[j].setAttribute('id', seats[flag].room+seats[flag].row + seats[flag].column);
                // currentCells[j].innerHTML = seats[flag].row + seats[flag].column;
                currentCells[j].innerHTML = " ";
                flag++;
            }
        }
    }
    document.getElementById(idName).appendChild(table);
}

```

3. Populate room layout dynamically

Step 1: Set initial status of seats

```

function seatClass(currentCells,j,condition){//get the status of seat
    switch (condition){
        case '-1':
            currentCells[j].setAttribute('class','noSeat');
            break;
        case '0':
            currentCells[j].setAttribute('class','showSeat');
            currentCells[j].setAttribute('onClick','seatSelect(this)');
            break;
        case '1':
            currentCells[j].setAttribute('class','bookedSeat');
            break;
    }
}

```

Step 2: Select a seat and cancel a seat


```

function seatSelect(cell){//change the style of selecting and canceling seats
    var currentClass=cell.getAttribute('class');
    if(allSelectedSeats.length==0){
        document.getElementById('disSeats').innerHTML='';
        document.getElementById('bookInfo').innerHTML='';
    }
    if(currentClass=="showSeat"){
        if(tickets>=6){
            alert("One order maximum can book 6 tickets!");
        }else{
            cell.setAttribute('class','selectSeat');
            tickets++;
            getValues(cell,1);
        }
        allSelectedSeats.push(cell.id);
    }
    if(currentClass=="selectSeat"){
        cell.setAttribute('class','showSeat');
        tickets--;
        getValues(cell,0);
        if(tickets==0){
            showSeats.style.borderColor="white";
        }
        var index = allSelectedSeats.indexOf(cell.id);
        if (index > -1) {
            allSelectedSeats.splice(index, 1);
        }
    }
}

```

4. Get booking information

1. Date, time and movie name are get from URL parameters.

2.Seat number and price

Step 1: Set Attributions

When dynamic creating seats, the detailed seat number and price are stored in the attributes of seatNumber and seatPrice.

```

currentCells[j].setAttribute('seatNumber', seats[flag].row + seats[flag].column);
currentCells[j].setAttribute('seatPrice', seats[flag].price);

```

Step 2: Get Attributions

When a user selects or cancels a seat, getting seat number and price from the attributes.

```

function getValues(cell,count){//set values for price and seats number
    if(count==1){
        var li= document.createElement("li");
        totalPrice+=parseInt(cell.getAttribute('seatPrice'));
        li.innerHTML=cell.getAttribute('seatNumber')+' $'+cell.getAttribute('seatPrice');
        li.setAttribute("id",cell.getAttribute('seatNumber'));
        showSeats.appendChild(li);
        showSeats.style.borderColor="orange";
    }else{
        totalPrice-=cell.getAttribute('seatPrice');
        var deletli=document.getElementById(cell.getAttribute('seatNumber'));
        showSeats.removeChild(deletli);
    }
    showPrice.innerHTML=' $ '+totalPrice;
    showTickets.innerHTML=" "+tickets;
}

```

3. Confirm booking

Step 1: Presenting a summary of booking information



Step 2: The selected seats are become unavailable.


```

function bookConfirmation(){
    var disAll='';
    if(allSelectedSearts.length==0){
        alert("Please select at least one seat!");
    }else{
        var seatInfo='';
        for (var i = 0; i < allSelectedSearts.length; i++) {
            document.getElementById(allSelectedSearts[i]).setAttribute('class','bookedSeat');
            if(roomName=="Puru"){
                allSelectedSearts[i]=allSelectedSearts[i].replace("Puru","");
                seatInfo+=allSelectedSearts[i]+' ';
            }else{
                allSelectedSearts[i]=allSelectedSearts[i].replace("Whero","");
                seatInfo+=allSelectedSearts[i]+' ';
            }
        }
        var bookInfo=document.getElementById("bookInfo");
        seatInfo=seatInfo.substr(0,seatInfo.length-1);
        disAll+="Movie: "+movieName;
        disAll+="<br/>Date: "+dateTime;
        disAll+="<br/>Room: "+roomName;
        disAll+="<br/>Seats: "+seatInfo+"<br/> Tickets: "+tickets+"<br/> TotalPrice: $ "+totalPrice;
        bookInfo.innerHTML=disAll;
        tickets=0;
        totalPrice=0;
        allSelectedSearts=[];
    }
}

```

• Test Plan

Requirement to test	Test Data Input	Expected Outcomes	Actual Outcomes
2.1 User should be able to select a date, time and cinema room (minimum two different movie sessions per day for each cinema room)	<ol style="list-style-type: none"> 1. Select a valid date "01/06/2019" 2. Select time "12:00" 3. Select movie "Dumbo" 4. Select "Whero Room" 	<ol style="list-style-type: none"> 1. Show "Whero Room" seats 2. Movie name, Room name, Date time are correct. 	<ol style="list-style-type: none"> 1. Show "Puru Room" seat 2. Movie name, Room name, Date time are correct.
2.2 User unable to input empty to a date.	<ol style="list-style-type: none"> 1. Leave date empty 2. Select time "14:30" 3. Select movie "Shazam!" 	<ol style="list-style-type: none"> 1. Show message "Please choose a date" 	<ol style="list-style-type: none"> 1. Show message "Please choose a date"
2.3 User unable to select a past date	<ol style="list-style-type: none"> 1. Select a past date "23/05/2019" 2. Select time "17:00" 3. Select movie "Hellboy" 	<ol style="list-style-type: none"> 1. Show message "Sorry, you cannot select a past date!" 	<ol style="list-style-type: none"> 1. Show message "Sorry, you cannot select a past date!"
3.1 Load cinema room settings from XML files.	<ol style="list-style-type: none"> 1. Select a valid date "23/06/2019" 2. Select time "9:30" 3. Select movie "Dumbo" 4. Click "Return" back to choose date, time, movie page 5. Change the xml name 6. Select a valid date "23/06/2019" 7. Select time "9:30" 	<ol style="list-style-type: none"> 1. First time successfully show "Puru Room" seats. 2. Second time failed to show "Puru Room" seats. 	<ol style="list-style-type: none"> 3. First time successfully show "Puru Room" seats. 4. Second time failed to show "Puru Room" seats.

	8. Select movie "Dumbo"		
4.1 Populate each room layout dynamically on the webpage. Layout for each room is correct as per requirements.	1. Select a valid date "24/06/2019" 2. Select time "9:30" 3. Select movie "Dumbo"	1. "Puru room" seats are showing correctly as requirement	1. "Puru room" seats are showing correctly as requirement
4.2 Change room, Layout for each room is correct as per requirements	1. Select a valid date "07/06/2019" 2. Select time "19:30" 3. Select movie "Shazam!" 4. Select "Whero Room" 5. Select "Puru Room"	1. "Puru room" seats are showing correctly as requirement 2. "Whero Room" seats are showing correctly as requirement	1. "Puru room" seats are showing correctly as requirement 2. "Whero Room" seats are showing correctly as requirement
4.3 User must be able to dynamically see row number, seat number and cost	1. Select a valid date "03/06/2019" 2. Select time "22:00" 3. Select movie "Shazam!" 4. Select seats A1 A2	1. When hover on the seat A1 A2 can see a tip show A1 \$20 A2 \$20 2. On the right can see 3. Seats A1 \$20 A2 \$20	1. When hover on the seat A1 A2 can see a tip show A1 \$20 A2 \$20 2. On the right can see 3. Seats A1 \$20 A2 \$20
4.4 Already booked seats clearly indicated in your layout and user should not be able to select them for booking.	1. Select a valid date "04/06/2019" 2. Select time "00:30" 3. Select movie "Five Feet Apart" 4. Click booked seats "D3 D4"	1. Unable to select booked seat.	1. Unable to select booked seat.
4.5 Seats from different rows must have different price range. Seats price for first two rows is \$20, for next three rows it is \$23, and rest of the rows are \$25.	1. Select a valid date "06/06/2019" 2. Select time "09:30" 3. Select movie "The LEGO Movie 2" 4. Select B1 C1 F1	1. Show price B1 \$20, C1 \$23, F1 \$25	1. Show price B1 \$20, C1 \$23, F1 \$25
5.1 If user wants to change their booking selection they system should be able to update dynamically.	1. Select a valid date "08/06/2019" 2. Select time "14:00" 3. Select movie "Dumbo" 4. Select B6,B7,B8,B9,B10 5. Cancel B6,B10	1. The status and price of the seats are dynamically update 2. After cancel B6,B10 only B7, B8, B9 are selected	1. The status and price of the seats are dynamically update 2. After cancel B6,B10 only B7, B8, B9 are selected
6.1 On user conformation the booking system should calculate price of dynamically. User should be able to see summary for booking with date, time, name of cinema room, seats numbers, cost for each seat and total	1. Select a valid date "11/06/2019" 2. Select time "19:30" 3. Select movie "Wonder Park1" 4. Select "Whero Room" 5. Select A4, C5, H5, H6	1. Show Movie : "Wonder Park1" 2. Room: Whero Room 3. Date: 2019-06-11 19:30 4. Seats: A4 \$20, C5 \$23, H5 \$25, H6 \$25 5. Tickets 4 6. Total price: \$93	1. Show Movie : "Wonder Park1" 2. Room: Whero Room 3. Date: 2019-06-11 19:30 4. Seats: A4 \$20, C5 \$23, H5 \$25, H6 \$25 5. Tickets 4 6. Total price: \$93

cost			
<p>7.1After booking conformation is done seats should be unavailable.</p>	<ol style="list-style-type: none"> 1. Select a valid date "21/06/2019" 2. Select time "14:30" 3. Select movie "Captain Marvel" 4. Select seat B1, B2 5. Click "BookNow" 6. Select B1,B2 	<ol style="list-style-type: none"> 1. After booking B1,B2 become gray seats which means unable to choose 2. Click B1,B2 unable to select 	<ol style="list-style-type: none"> 1. After booking B1,B2 become gray seats which means unable to choose 2. Click B1,B2 unable to select
<p>7.2 After booking when come back to the same date, time, movie and room the booked seats are still showing booked status</p>	<ol style="list-style-type: none"> 1. Select a valid date "07/06/2019" 2. Select time "17:00" 3. Select movie "Avengers:Endgame" 4. Select seat E6, E7 5. Click "BookNow" 6. Click "Return" back to the page to choose date "08/06/2019" time "17:00" movie "Captain Marvel" 7. Select seats E6,E7 8. Cancel E6, E7 9. Click "Return" back to the page to choose date "07/06/2019" time "17:00" movie "Captain Marvel" 	<ol style="list-style-type: none"> 1. When second time return to the Puru Room seats "07/06/2019" time "17:00" movie "Captain Marvel" E6,E7 still shows booked status 	<ol style="list-style-type: none"> 1. When second time return to the Puru Room seats "07/06/2019" time "17:00" movie "Captain Marvel" E6,E7 still shows booked status