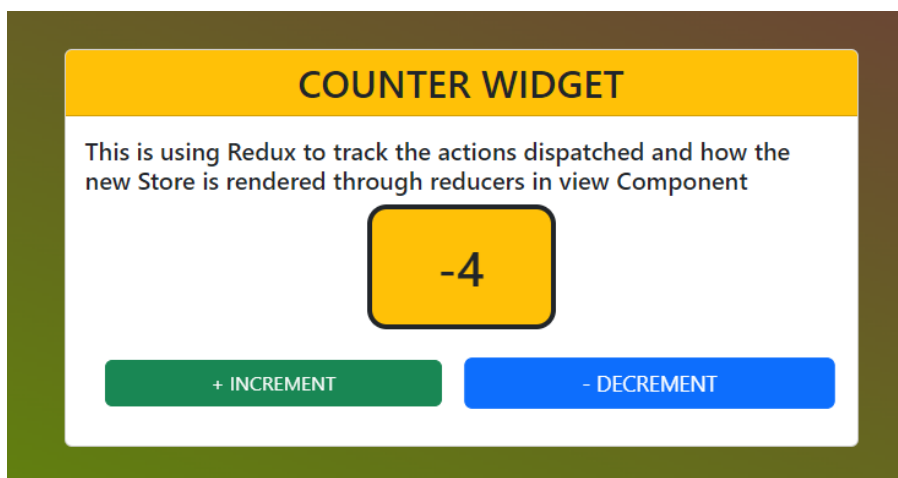
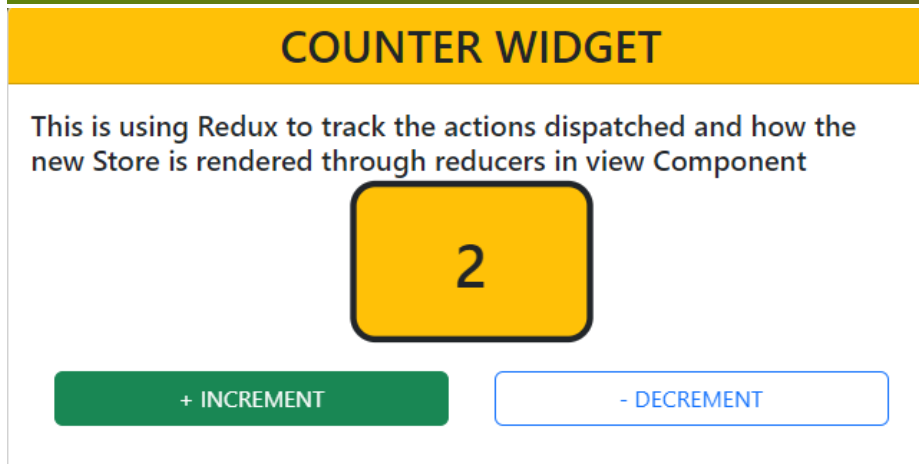
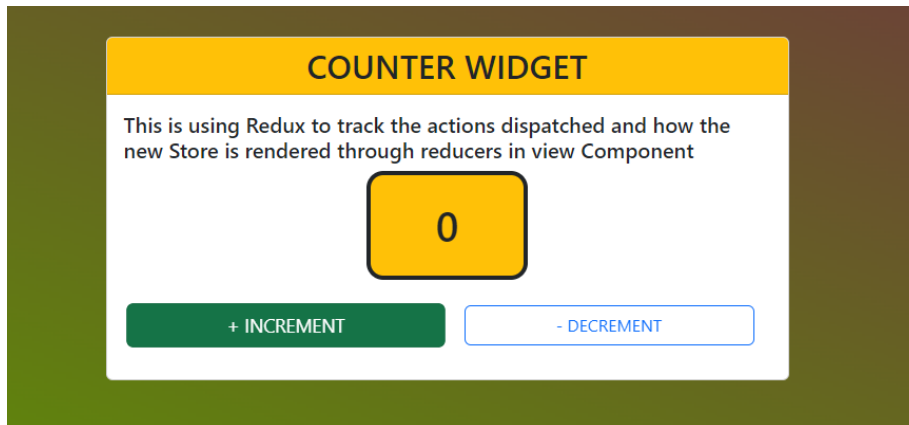


COUNTER WIDGET APPLICATION

Using HTML, CSS, JavaScript, Bootstrap, Redux



HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Counter using JS + Redux</title>
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css"
rel="stylesheet">
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"></script>

  <script
src="https://cdnjs.cloudflare.com/ajax/libs/redux/4.2.1/redux.min.js"
integrity="sha512-
1/8Tj23BRrWnKZXeBruk6wTnsMJbi/lJsk9bsRgVwb6j5q39n0A00gFjbCTaDo5l5XrPVv4DZXftrJ
ExhRF/Ug==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <link rel="stylesheet" href="style.css">
</head>
<body>

  <div class="container p-5">
    <div class="card w-75 ">
      <h2 class="card-header bg-warning text-dark text-center text-uppercase
font-weight-bolder ">Counter Widget</h2>
      <div class="card-body">
        <h5 class="card-title text-justify ">This is using Redux to track
the actions dispatched and how the new Store is rendered through reducers in
view Component</h5>

        <h1 id="counterH1" class="container bg-warning p-4 rounded-4 mx-6 w-
25 text-center border border-4 border-dark hvr-grow "> </h1>

        <div class="d-flex align-items-center justify-content-center ">
          <button id="incr" class="btn hvr-grow btn-success m-3 w-100" >+
INCREMENT</button>
          <button id="decr" class="btn hvr-grow btn-outline-primary m-3 w-
100 ">- DECREMENT</button>
        </div>
      </div>
    </div>
  </div>

<script>
  var counterH1=document.getElementById('counterH1')
```

```
var incrBtn=document.getElementById('incr')
var decrBtn=document.getElementById('decr')

var incrAction={type:'INCR'}
var decrAction={type:'DECR'}

function counterReducer(state,action){
  if(typeof state ==='undefined') return 0
  if(action.type==='INCR'){return state + 1}
  else if(action.type==='DECR'){return state -1}
  else {return state }
}

const store=Redux.createStore(counterReducer)

function render(){
  counterH1.innerHTML=store.getState()
}

render()
store.subscribe(render)

incrBtn.onclick = function(){
  store.dispatch(incrAction)
}
decrBtn.onclick = function(){
  store.dispatch(decrAction)
}

</script>

</body>
</html>
```

CSS

```
body{  
    background: linear-gradient(to top right, rgb(96, 132, 13), rgb(109, 60, 60));  
    background-repeat: no-repeat;  
    background-position: right top;  
}  
.hvr-grow {  
  
    transform: translateZ(0);  
    box-shadow: 0 0 1px rgba(0, 0, 0, 0);  
    backface-visibility: hidden;  
    -moz-osx-font-smoothing: grayscale;  
    transition-duration: 0.3s;  
    transition-property: transform;  
}  
  
.hvr-grow:hover,  
.hvr-grow:focus,  
.hvr-grow:active {  
    transform: scale(1.1);  
}
```