

Experiment 2

AIM: Responsive Banking UI with Deposit/Withdrawal Buttons and Balance Display

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8"/>
  <meta name="viewport" content="width=device-width, initial-
scale=1.0"/>
  <title>Banking UI</title>
  <link
href="https://fonts.googleapis.com/css2?family=Poppins&displa
y=swap" rel="stylesheet">
  <style>
    body {
      font-family: 'Poppins', sans-serif;
      background: #e6f0ff;
      margin: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }

    .bank-container {
      background: white;
      padding: 30px;
      border-radius: 12px;
      box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
      text-align: center;
      max-width: 400px;
      width: 100%;
    }

    h2 {
      margin-bottom: 20px;
```

```
}

.balance {
  font-size: 2.5rem;
  font-weight: bold;
  color: #333;
  margin-bottom: 30px;
}

.actions button {
  padding: 12px 20px;
  font-size: 1rem;
  margin: 5px;
  border: none;
  border-radius: 8px;
  cursor: pointer;
  transition: 0.2s ease;
}

.actions button:nth-child(1) {
  background: #28a745;
  color: white;
}

.actions button:nth-child(2) {
  background: #dc3545;
  color: white;
}

.actions button:hover {
  opacity: 0.9;
}
</style>
</head>
<body>
  <div class="bank-container">
    <h2>My Bank Account</h2>
    <div class="balance">${<span id="balance">1000</span></div>
    <div class="actions">
      <button onclick="deposit()">Deposit $100</button>
```

```
<button onclick="withdraw()">Withdraw $100</button>
</div>
</div>

<script>
let balance = 1000;
const balanceDisplay = document.getElementById("balance");

function deposit() {
  balance += 100;
  balanceDisplay.textContent = balance;
}

function withdraw() {
  if (balance >= 100) {
    balance -= 100;
    balanceDisplay.textContent = balance;
  } else {
    alert("Insufficient funds!");
  }
}
</script>
</body>
</html>
```

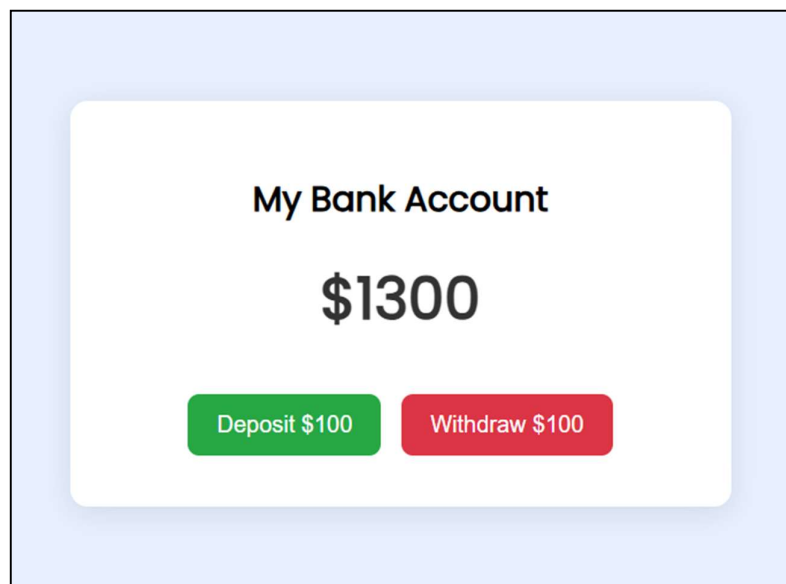


Figure 1: Experiment 2 Banking UI output