

## Web Technology Assignment-9

---

### T1: Code in Vue.js

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Currency Converter</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #222;
      color: white;
    }
    #app {
      max-width: 400px;
      margin: 0 auto;
      padding: 20px;
      border: 1px solid #ccc;
      border-radius: 5px;
      background-color: #333;
    }
    input[type="number"], select, button {
      margin-bottom: 10px;
      padding: 5px;
      border: 1px solid #ccc;
      border-radius: 3px;
      background-color: #444;
      color: white;
    }
    button {
      cursor: pointer;
      background-color: #007bff;
      color: white;
      border: none;
    }
    p {
```

```

    font-size: 18px;
  }
</style>
</head>
<body>
  <div id="app">
    <h2>Currency Converter</h2>
    <div>
      <label>
        Amount:
        <input type="number" v-model="amount" />
      </label>
    </div>
    <div>
      <label>
        From:
        <select v-model="fromCurrency">
          <option value="USD">USD</option>
          <!-- Add other currency options here -->
        </select>
      </label>
    </div>
    <div>
      <label>
        To:
        <select v-model="toCurrency">
          <option value="INR">INR</option>
          <!-- Add other currency options here -->
        </select>
      </label>
    </div>
    <button @click="convertCurrency">Convert</button>
    <div v-if="convertedAmount">
      <p>{{ amount }} {{ fromCurrency }} = {{ convertedAmount }} {{ toCurrency }}</p>
    </div>
  </div>

  <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
  <script>
    new Vue({
      el: '#app',

```

```
data: {  
  amount: '',  
  fromCurrency: 'USD',  
  toCurrency: 'INR',  
  convertedAmount: null,  
  exchangeRate: 75.0 // Hard-coded exchange rate (1 USD = 75 INR)  
},  
methods: {  
  convertCurrency() {  
    this.convertedAmount = (this.amount * this.exchangeRate).toFixed(2);  
  }  
}  
});  
</script>  
</body>  
</html>
```

Web Page:

The screenshot shows a web browser window with a tab titled 'Web View'. The page content is a 'Currency Converter' form. It includes an input field for 'Amount' with the value '45', a 'From' dropdown menu set to 'USD', and a 'To' dropdown menu set to 'INR'. A blue 'Convert' button is positioned below the dropdowns. The result of the conversion is displayed as '45 USD = 3375.00 INR'. The browser's address bar and navigation icons are visible at the top, and a help icon (?) is in the bottom right corner.

## T2: Code in Vue.js

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Stopwatch</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      color: white;
      background-color: #222;
    }
    .container {
      text-align: center;
      margin-top: 50px;
    }
    .button {
      margin: 5px;
      padding: 10px 20px;
      font-size: 16px;
      cursor: pointer;
      background-color: #007bff;
      color: white;
      border: none;
      border-radius: 5px;
    }
    .button:hover {
      background-color: #0056b3;
    }
    .button:active {
      background-color: #004080;
    }
  </style>
</head>
<body>
  <div id="app" class="container">
    <h2>Stopwatch</h2>
    <div>{{ formatTime }}</div>
    <div>
      <button class="button" @click="startStopwatch" v-
if="!running">Start</button>
```

```

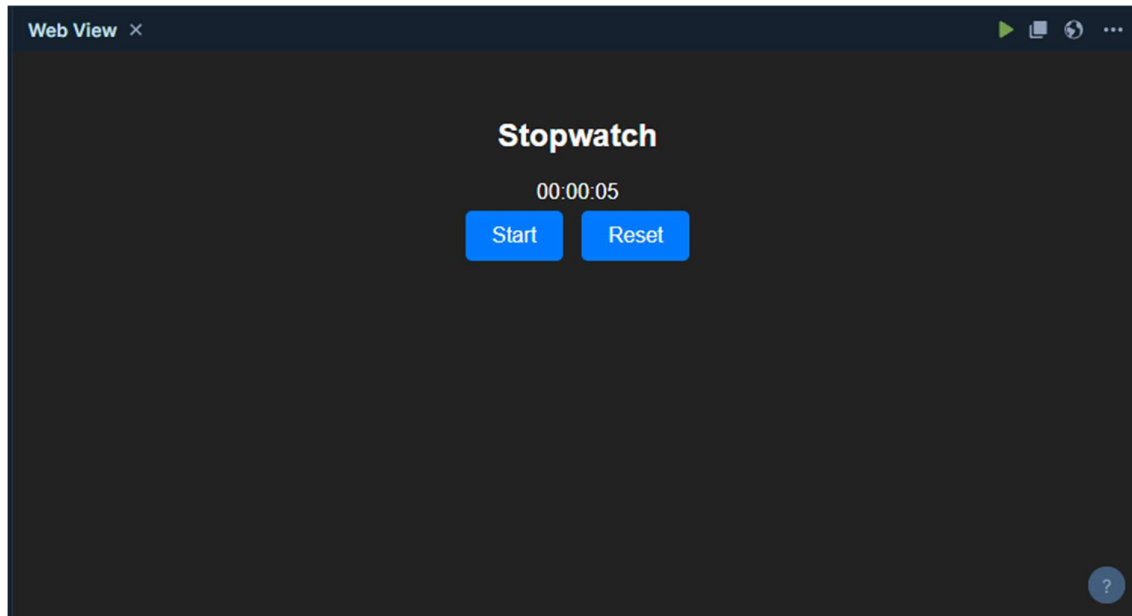
        <button class="button" @click="pauseStopwatch" v-
if="running">Pause</button>
        <button class="button" @click="resetStopwatch">Reset</button>
    </div>
</div>

<script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
<script>
    new Vue({
      el: '#app',
      data: {
        running: false,
        time: 0,
        timer: null
      },
      computed: {
        formatTime() {
          const hours = Math.floor(this.time / 3600);
          const minutes = Math.floor((this.time % 3600) / 60);
          const seconds = this.time % 60;
          return `${hours.toString().padStart(2,
'0')}:${minutes.toString().padStart(2, '0')}:${seconds.toString().padStart(2,
'0')}`;
        }
      },
      methods: {
        startStopwatch() {
          this.running = true;
          this.timer = setInterval(() => {
            this.time++;
          }, 1000);
        },
        pauseStopwatch() {
          clearInterval(this.timer);
          this.running = false;
        },
        resetStopwatch() {
          clearInterval(this.timer);
          this.running = false;
          this.time = 0;
        }
      }
    })

```

```
});  
</script>  
</body>  
</html>
```

## Web Page:



## T3: code in vue.js

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Messaging Application</title>  
<style>  
  body {  
    font-family: Arial, sans-serif;  
    background-color: #f0f0f0;  
    margin: 0;  
    padding: 0;  
  }  
  .container {  
    display: flex;
```

```
    justify-content: space-between;
    height: 100vh;
  }
  .user {
    flex: 1;
    padding: 20px;
    box-sizing: border-box;
    background-color: #fff;
    border: 1px solid #ccc;
    overflow-y: auto;
  }
  .messages {
    margin-bottom: 10px;
  }
  .message {
    background-color: #eaeaea;
    padding: 5px 10px;
    border-radius: 5px;
    margin-bottom: 5px;
  }
  textarea {
    width: 100%;
    padding: 10px;
    box-sizing: border-box;
    margin-bottom: 10px;
  }
  button {
    padding: 10px 20px;
    background-color: #007bff;
    color: white;
    border: none;
    border-radius: 5px;
    cursor: pointer;
  }
  button:hover {
    background-color: #0056b3;
  }
</style>
</head>
<body>
  <div id="app" class="container">
    <div class="user" id="userA">
```

```

    <h2>User A</h2>
    <div class="messages">
      <div v-for="message in userAMessages" class="message">Message from A:
{{ message }}</div>
    </div>
    <textarea v-model="userAInput" placeholder="Type your
message"></textarea>
    <button @click="sendMessage('userB', userAInput)">Send</button>
  </div>
  <div class="user" id="userB">
    <h2>User B</h2>
    <div class="messages">
      <div v-for="message in userBMessages" class="message">Message from B:
{{ message }}</div>
    </div>
    <textarea v-model="userBInput" placeholder="Type your
message"></textarea>
    <button @click="sendMessage('userA', userBInput)">Send</button>
  </div>
</div>

<script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
<script>
  new Vue({
    el: '#app',
    data: {
      userAInput: '',
      userAMessages: [],
      userBInput: '',
      userBMessages: []
    },
    methods: {
      sendMessage(receiver, message) {
        this[receiver + 'Messages'].unshift(message);
        this[receiver + 'Input'] = ''; // Clear input field after sending
message
      }
    }
  });
</script>
</body>
</html>

```



Web Page:

Web View

User A

Message from A: hello iam B

how are you

Send

User B

Message from B: how are you

Message from B: hello iam user A

Type your message

Send