

# Industrial Training Daily Diary

## MERN Stack Training

**Company:** Sensation Software Solutions

**Student Name:** Gurvinder Singh

**Training Duration:** 6 Months

**Day:** 19

---

## Day 19 – Redux Advanced: Middleware, Async Actions & Redux Thunk

### Objective of the Day

The objective of Day 19 was to understand **advanced Redux concepts**, including **middleware**, **asynchronous actions**, and how to handle them using **Redux Thunk** for making API calls and managing side effects in a React-Redux application.

---

### Introduction to Middleware in Redux

The trainer explained that **middleware** provides a way to extend Redux with custom functionality. Middleware can intercept actions before they reach the reducer, allowing developers to:

- Log actions
  - Perform asynchronous tasks
  - Modify actions or state
- 

### Asynchronous Actions in Redux

We learned that normal Redux actions are **synchronous**, which limits the ability to handle API calls or async operations directly.

Key points:

- Async actions cannot be handled by default Redux
  - Middleware like Redux Thunk is used to manage async behavior
-

# Redux Thunk

**Redux Thunk** allows us to write action creators that return a **function instead of an action**. This function can perform asynchronous tasks and dispatch actions when operations complete.

Practical points covered:

- Installing Redux Thunk
  - Configuring middleware in the store
  - Writing async action creators
  - Dispatching actions after API call responses
- 

## Practical Demonstration

During lab session:

- Configured Redux store with Redux Thunk middleware
- Created async actions to fetch data from a mock API
- Used `useDispatch` and `useSelector` hooks to integrate with React components
- Implemented loading and error handling states

This hands-on session helped understand real-world async state management.

---

## Weekly Task – Week 5 (Assigned)

The weekly task included:

- Extend the React-Redux project
  - Implement async API calls using Redux Thunk
  - Manage loading, success, and error states
  - Update diary and push code to GitHub Pages
- 

## Importance in MERN Stack Development

Middleware and async action management are crucial for handling backend API calls efficiently, ensuring the frontend application is responsive and robust.

---

## Conclusion of Day 19

Day 19 enhanced my understanding of advanced Redux concepts, enabling me to handle asynchronous operations and integrate middleware effectively in React-Redux applications.

---

**Outcome of the Day:**

- Understanding of Redux middleware
- Ability to implement async actions using Redux Thunk
- Hands-on experience with API calls