

# Summer Internship Report at Extech Digital

Presented by: Sidhant Kapoor (2021IMT-100)

Supervisor: Prof. Shashikala Tapaswi

ABV-Indian Institute of Information Technology and Management Gwalior



# Table of Contents

- 1 Company Profile
- 2 Problem Statement
- 3 Objectives
- 4 Literature Review
- 5 Solution Approach
- 6 Results
- 7 References

# Company Profile - Extech Digital



Figure 1: Extech Digital Logo

- **Overview:** Extech Digital is a dynamic outsourcing company based in Mohali, specializing in delivering comprehensive services to meet the diverse needs of its clients. With expertise spanning everything from testing and maintenance to full-fledged development.
- **Mission:** Drive client business growth through innovative web design, development, and marketing solutions. Deliver high-quality, impactful services that provide clients with a competitive edge in the global market.
- **Vision:** Become a leading global digital marketing partner, known for delivering creative and effective solutions that help businesses thrive in the ever-evolving digital landscape. Aim to be the go-to agency for achieving online success.

# Problem Statement

## Key Challenges in the Existing Platform

- **Lack of Reusable UI Components** – The existing web platform lacked reusable UI components, leading to inconsistent design and increased development time.
- **Inefficient Data Export Methods** – Absence of efficient data export methods made it difficult for users to manage and analyze reports effectively.
- **Limited Integrated Tools** – No integrated financial tools or content management systems, restricting functionality and reducing user engagement.

# Objectives

- **Develop a Robust & Scalable Platform**

- Build a strong, reliable, and user-friendly system
- Ensure easy maintenance and future scalability

- **Improve UI Consistency & Data Handling**

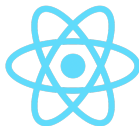
- Resolve issues with inconsistent UI components
- Implement efficient data handling methods
- Enhance overall user experience and system efficiency

- **Deliver Seamless User & Admin Experience**

- Provide smooth and intuitive usage for end-users
- Simplify administration and management tasks
- Meet the growing demands and expectations

# Literature Review

## React



- **JavaScript Library for UI:** A powerful JavaScript library for building dynamic and interactive user interfaces.
- **Component-Based Development:** Enables component-based development, making applications more modular, reusable, and maintainable.
- **Virtual DOM for Performance:** Virtual DOM efficiently updates UI elements, improving performance and responsiveness.
- **Ecosystem & Community Support:** Strong ecosystem and community support make it ideal for developing scalable and modern web applications.

# Literature Review

## MongoDB



- **NoSQL Database:** A leading NoSQL database designed for flexibility and scalability.
- **Document-Based Storage:** Stores data in JSON-like documents, making it easy to handle unstructured or semi-structured data.
- **High Performance & Scalability:** Provides high performance, horizontal scaling, and powerful querying features.
- **Used in Modern Applications:** Widely used for modern applications requiring big data, real-time analytics, and distributed systems.

# Literature Review

## Storybook



- **Open-Source UI Tool:** An open-source tool for developing and testing UI components in isolation.
- **Reusable & Consistent Components:** Helps create reusable, consistent, and well-documented components.
- **Collaboration Between Teams:** Enhances collaboration between developers and designers with a shared component library.
- **Better Testing & Scalability:** Improves testing, visualization, and scalability of component-driven development.



# Literature Review

## Express.js & Node.js



- **Node.js:** A JavaScript runtime built on Chrome's V8 engine, enabling fast, scalable, and event-driven server-side applications.
- Supports non-blocking I/O operations, making it highly efficient for real-time applications.
- **Express.js:** A lightweight web framework for Node.js that simplifies building APIs and server-side applications.
- Provides robust routing, middleware support, and flexibility for scalable web app development.

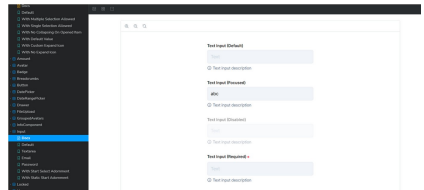
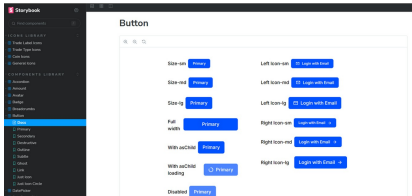
# Solution Approach

## Task I: UI Components Development with Storybook

- Developed a suite of reusable UI components using Storybook, ensuring modularity and reusability.
- Created fundamental components like buttons, accordions, date pickers, currency inputs, and menus with multiple variants for different use cases.
- Leveraged Storybook for isolated development, facilitating rigorous testing and real-time feedback.
- Ensured components were adaptable, supporting diverse design specifications and user interactions.
- Streamlined the development process, enhancing frontend efficiency and maintaining a cohesive user interface.

# Solution Approach

## Task I: UI Components Development (Demo)



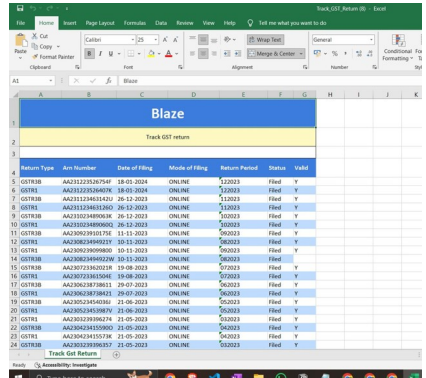
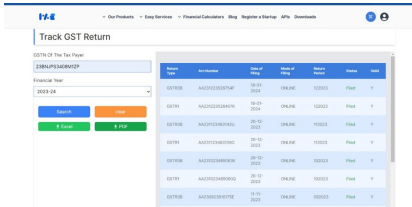
# Solution Approach

## Task II: Excel Export Functionality

- Developed a user-friendly Excel export feature for tax reports using Sheet.js andxlsx libraries.
- Implemented custom styles for headers, rows, and columns to ensure data accuracy and visual consistency with the application's UI.
- Created utility functions for data transformation, cell formatting, and workbook creation, leading to cleaner, modular code.
- Rigorously tested the feature for compatibility with different Excel versions and large datasets.

# Solution Approach

## Task II: Excel Export Functionality (Demo)



# Solution Approach

## Task III: Financial Calculators

- Developed financial calculators using React Hook Form, and Chart.js to provide accurate and user-friendly computations.
- Utilized React Hook Form for efficient form management, minimizing re-renders and ensuring easy maintenance.
- Incorporated Chart.js for dynamic visualizations, creating intuitive charts (line, bar, pie) that made complex financial data more accessible.
- Conducted rigorous testing to ensure accuracy and reliability, delivering a polished feature for informed financial decision-making.
- Focused on modularity and reusability, enabling easy customization and future updates.

# Solution Approach

## Task III: Financial Calculators (Demo)

**TAX CALCULATOR**

PAN No. \*

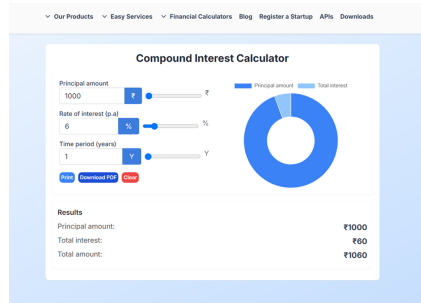
Tax Payer \*

Status

First Name

Last Name

Assessment Year \*



# Solution Approach

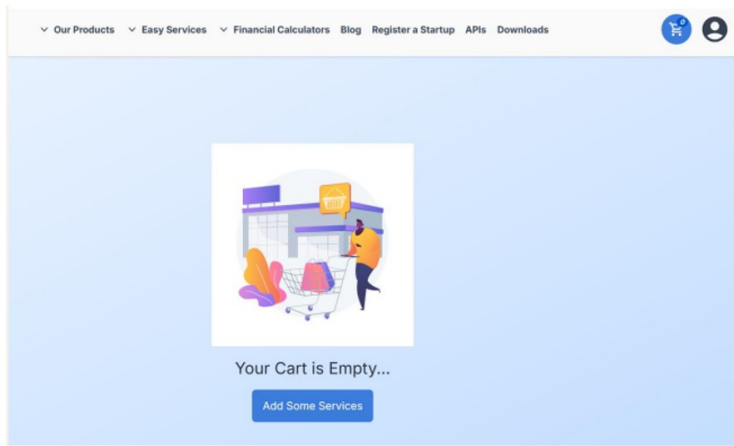
## Task IV: Cart Management System

- Engineered a Cart Management System using React for efficient backend API routes and Redux for centralized state management.
- Utilized Redux with middleware to handle asynchronous actions, ensuring real-time updates and smooth user interactions.
- Implemented local storage for cart state persistence, enhancing user experience by maintaining cart continuity across sessions.
- Applied memoization to optimize performance, reducing unnecessary re-renders and ensuring responsiveness even with complex cart operations.



# Solution Approach

## Task IV: Cart Management System (Demo)



# Result

- **Enhanced Technical Expertise:** Improved skills with technologies like React.js, Next.js, MongoDB, and Redux, contributing to the development of scalable and efficient web applications.
- **Full-Stack Development:** Gained comprehensive experience in both frontend and backend development, including API integration, state management, and server-side rendering.
- **Performance Optimization:** Implemented caching, memoization, and asynchronous data handling techniques to enhance application performance and user experience.
- **Collaboration & Communication:** Worked closely with cross-functional teams, improving collaboration, and contributing to successful project outcomes.

# References

- [1] MongoDB. MongoDB Documentation. <https://docs.mongodb.com/>
- [2] Express.js. Express.js Documentation. <https://expressjs.com/>
- [3] React. React.js Documentation.  
<https://reactjs.org/docs/getting-started.html>
- [4] Node.js. Node.js Documentation. <https://nodejs.org/en/docs/>
- [5] Bootstrap. Bootstrap Documentation.  
<https://getbootstrap.com/docs/5.1/getting-started/introduction/>
- [6] Jest. Jest Documentation. <https://jestjs.io/docs/getting-started>
- [7] Mocha. Mocha Documentation. <https://mochajs.org/>
- [8] SeleniumHQ. Selenium Documentation.  
<https://www.selenium.dev/documentation/en/>
- [9] Auth0. What is JWT (JSON Web Token)?.  
<https://auth0.com/learn/json-web-tokens/>

Thank you!