Day 1: CONCEPTUALIZATION AND MARKETPLACE DESIGN

- Find the business vision for the WAY MORE marketplace
- Make a roadmap for the marketplace how the things are working
- Select our targeted audience for our selected product

Day2: TECHNICAL PLANNING.

- Successfully implemented Next.js to develop a high-performance, server-rendered e-commerce platform.
- Integrated the Sanity API as the content management backend for flexible and scalable data processing.
- Utilized Tailwind CSS to create a responsive and efficient UI with a utility-first methodology.
- Designed the system architecture to facilitate seamless API interactions and third-party integrations.
- Developed a technical roadmap to enhance the store's performance, scalability, and overall design.

Day3: DATA MIGRATION.

- Fetch the data from the API and then add it into the Sanity CMS through the data migration
- Fetch the data from the Sanity CMS and then display it on the front-end

Day 4: Dynamic Frontend Components.

- Make a dynamic page in the next.js project and then show the product detail of all product dynamically
- Make a suggested product component that show the product in the dynamic page of same category
- Make filter of category that will show the product of the same category
- Make add to cart function using react context api that adds the product in the cart page.

Day 5 : TESTING AND BACKEND REFINEMENTS.

- Performed extensive testing of the e-commerce platform, covering functionality, performance, and user experience.
- Adjusted Sanity API settings to enhance data flow and maintain content precision.
- Evaluated Next.js server-side rendering and dynamic routing to ensure smooth navigation and fast page loading.
- Enhanced backend API responses and verified seamless integration with frontend components styled in Tailwind CSS.
- Resolved bugs, optimized code, and ensured the backend adhered to scalability and security best practices.

Day 6 - Deployment Preparation and Staging Environment Setup

- Configured a staging environment replicating the production setup for testing and quality assurance.
- Deployed the Next.js application on a temporary hosting service to mimic real-world scenarios.
- Connected the Sanity API and ensured smooth data flow and content management in the staging environment.
- Evaluated Tailwind CSS-styled components for responsiveness and compatibility across different devices.
- Performed final pre-launch assessments, including performance tuning and security checks.