# Day-2 Progress Documentation

## Introduction

This document provides an overview of the technical workflow of our website, integrating a Sanity CMS backend, a Next.js frontend, TailwindCSS for styling, and REST API for data fetching. This workflow ensures a seamless and efficient user experience while maintaining robust content management capabilities.

## Technical Workflow

### Frontend: Next.js

Next.js is utilized for building the frontend of our website. It provides server-side rendering (SSR) and static site generation (SSG) capabilities, enabling faster page loads and better SEO performance. The dynamic routing feature in Next.js ensures a modular and scalable codebase.

### Styling: TailwindCSS

TailwindCSS is used for styling the frontend components. It offers a utility-first CSS framework, allowing rapid UI development with minimal custom CSS. TailwindCSS ensures consistency across the design system.

### Backend: Sanity CMS

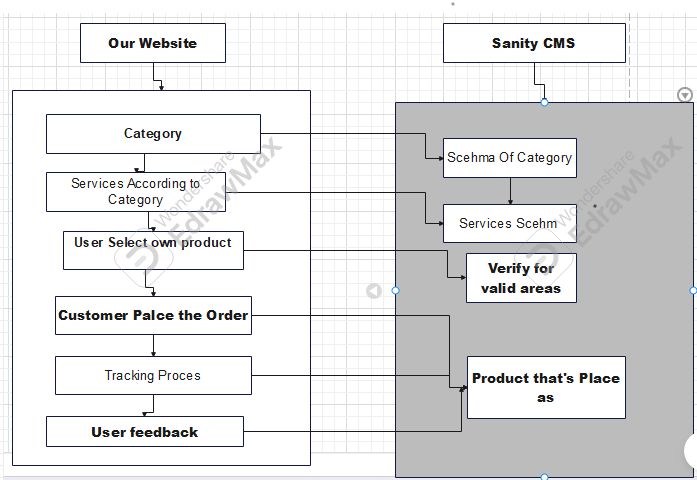
Sanity CMS serves as the backend content management system. It provides an intuitive interface for managing schemas, categories, and services, ensuring data is structured and validated before being presented on the website.

### Data Fetching: REST API

REST API is implemented to facilitate communication between the frontend and backend. APIs are used to fetch data from Sanity CMS and display it dynamically on the website.

## Workflow Diagram

The following diagram outlines the workflow between the website frontend and the Sanity CMS backend:

1. Users access the website and browse through various categories.  
2. Based on the selected category, services and products are displayed.  
3. Users select their desired product and place an order.  
4. The order information is tracked and logged, allowing for updates and feedback.  
5. On the backend, Sanity CMS manages categories and services schemas, verifies valid data, and ensures the product is placed accurately in the appropriate category.  
  
  
  
  
Perpared by: Muhammad Emad Hassan