PERPARED BY SEHER ZAHIDA

Day 2
Planning The Technical
Foundation

Day 2: Assignment Title: Furniture E-Commerce Website - Assignment

1. Introduction

Explain the purpose of your e-commerce website. Here's an example:
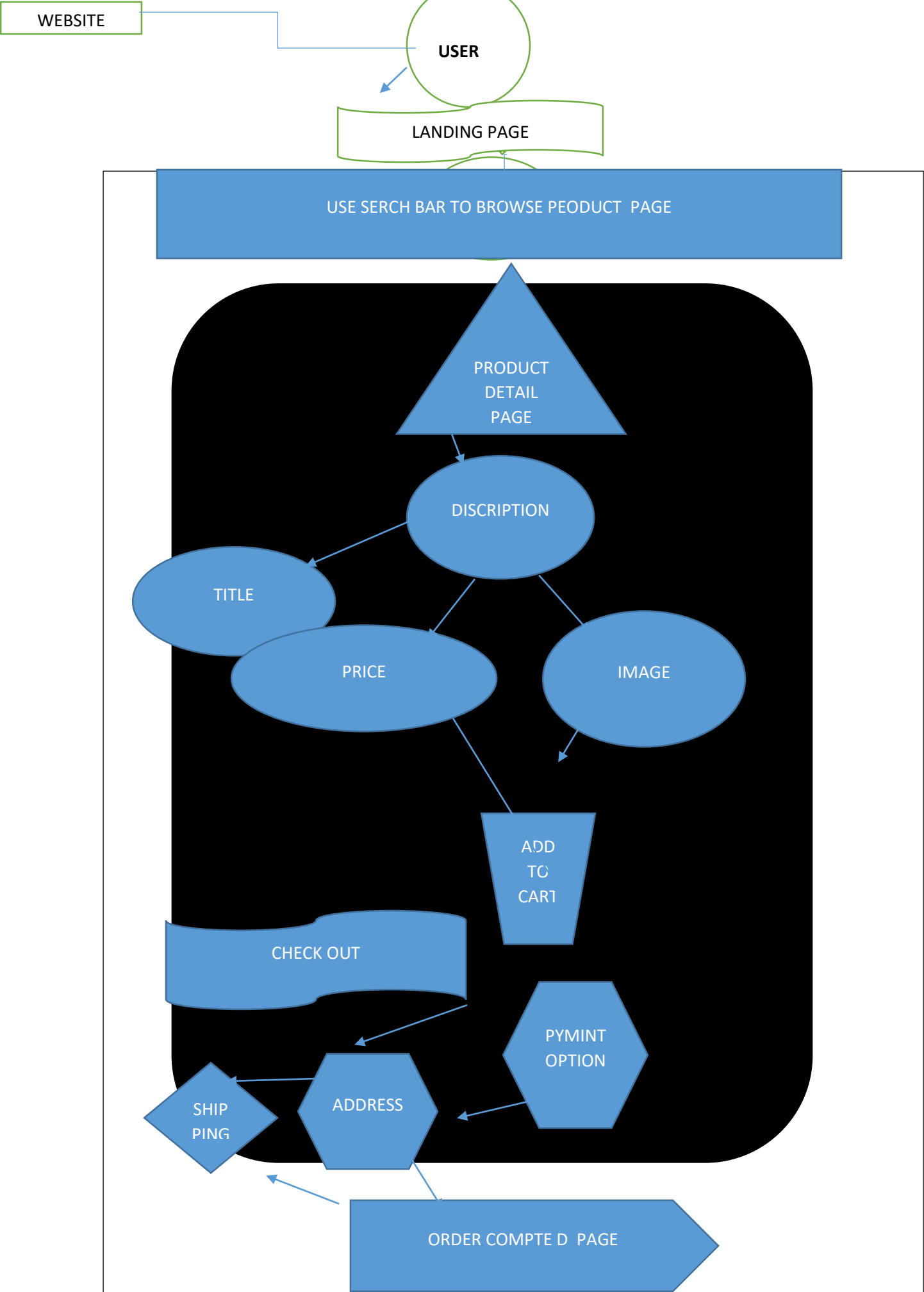
"This document outlines the technical and functional aspects of a furniture e-commerce website, focused on selling chairs. The goal is to create a user-friendly platform with seamless shopping experiences, efficient system architecture, and robust APIs.

No 2:

(WORK FLOW DIAGRAM)

WEBSITE

USER

LANDING PAGE

USE SERCH BAR TO BROWSE PEODUCT PAGE

PRODUCT DETAIL PAGE

DISCRIPTION

TITLE

PRICE

IMAGE

ADD TO CART

CHECK OUT
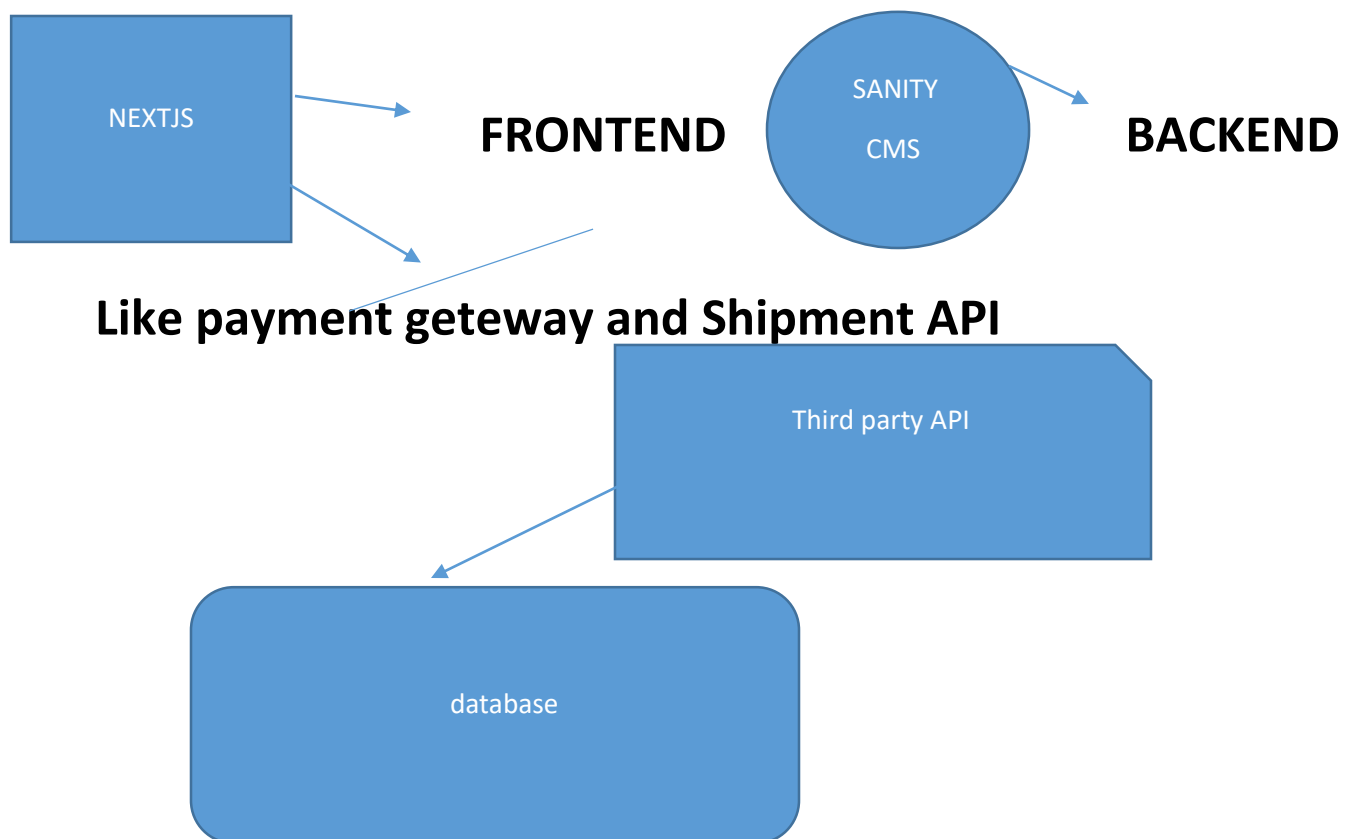
PYMINT OPTION

SHIP PING

ADDRESS

ORDER COMPTE D PAGE

"The work flow diagram illustrates the user journey,starting from visiting the website to completing a purchase."

3. System Architecture Diagram

A system architecture diagram shows how the frontend, backend (Sanity CMS), and APIs work together.

(SYSTEM ARCHTECTURE DIAGRAM)

NEXTJS

FRONTEND

SANITY

CMS

BACKEND

Like payment geteway and Shipment API

Third party API

database

Prepare by seher zahida

Diagram Description:

"This diagram illustrates how the website interacts with backend services (Sanity CMS) and APIs to provide product data, process orders, and manage payments

4. Conclusion

"The assignment covers the essential technical foundation of a furniture e-commerce website. The workflow, system architecture, and APIs ensure seamless functionality, making the website scalable and user-friendly."

Prepare by seher

SanitySchemaTypes.js

```
export default {
 name: 'product',
 title: 'Product',
 type: 'document',
 fields: [
  {
    name: 'name',
    title: 'Product Name',
```

```
      type: 'string',
      validation: (Rule) => Rule.required().min(2).max(100),
    },
    {
      name: 'slug',
      title: 'Slug',
      type: 'slug',
      options: {
        source: 'name',
        maxLength: 200,
      },
      validation: (Rule) => Rule.required(),
    },
    {
        name: 'price',
    title: 'Price',
      type: 'number',
      validation: (Rule) => Rule.required().min(0),
    },
    {
      name: 'image',
      title: 'Product Image',
      type: 'image',
      options: {
```

```
      hotspot: true,
    },
    fields: [
     {
       name: 'alt',
       title: 'Alternative Text',

       type: 'string',
       options: {
         isHighlighted: true,
       },
     },
     ],
    name: 'description',
  title: 'Description',
    type: 'text',
    validation: (Rule) => Rule.required().min(10).max(2000),
   },
  ],
};
```