This document outlines the design for a straightforward, scalable, and secure e-commerce website using **WordPress** with the **WooCommerce** plugin, hosted on the **Microsoft Azure** cloud platform.

**1. Goals and Objectives 🎯**

* **Primary Goal:** To create a functional and user-friendly online store to sell products directly to consumers.
* **Key Objectives:**
  + Ensure a secure and seamless checkout experience.
  + Provide an easy-to-use admin interface for managing products, orders, and inventory.
  + Build a scalable platform that can handle fluctuations in user traffic.
  + Maintain fast page load times for an optimal user experience.

**2. Technology Stack 🛠️**

* **Frontend & CMS:** **WordPress**.
  + **E-commerce Engine:** **WooCommerce** plugin for all shop functionalities (products, cart, checkout).
  + **Theme/Builder:** A lightweight theme (e.g., Astra) with a page builder like **Elementor** for easy customization.
* **Backend Hosting & Infrastructure:** **Microsoft Azure**.
  + **Web Hosting:** **Azure App Service for Linux** (WordPress on App Service template). This provides a managed, scalable environment for the WordPress site.
  + **Database:** **Azure Database for MySQL - Flexible Server**. A managed relational database service for WordPress data.
  + **Media Storage (Recommended):** **Azure Blob Storage** for storing product images and other media files to offload the web server and improve performance.
* **Payment Gateway:** **Stripe** and/or **PayPal**, integrated via official WooCommerce plugins.

**3. System Architecture**

The architecture is designed for simplicity and scalability. The user interacts directly with the WordPress application, which handles all business logic and data persistence.

**Simplified Data Flow:**

1. **Customer:** Accesses the webshop via a browser.
   * Browser -> Azure App Service (WordPress/WooCommerce) -> Azure Database for MySQL
2. **Checkout:** The customer proceeds to checkout.
   * WooCommerce -> Stripe/PayPal API (for payment processing) -> Order Creation in Azure DB
3. **Administrator:** Manages the store via the WordPress dashboard.
   * Admin Browser -> WordPress Admin (/wp-admin) -> Read/Write to Azure DB

**4. Key Features ✨**

**Customer-Facing Features**

* **Product Catalog:** A browsable grid of all products with sorting and filtering capabilities (e.g., by category, price).
* **Product Detail Pages:** Individual pages for each product with descriptions, image galleries, price, and an "Add to Cart" button.
* **Shopping Cart:** A persistent cart where users can view, modify, and manage their selected items.
* **Secure Checkout:** A multi-step form to collect shipping information and process payments securely.
* **User Accounts (Optional):** Ability for customers to register, log in, and view their order history.

**Administrator-Facing Features (via /wp-admin)**

* **Product Management:** A CRUD (**C**reate, **R**ead, **U**pdate, **D**elete) interface for products, including pricing, images, and categories.
* **Order Management:** A dashboard to view incoming orders, update their status (e.g., processing, shipped), and contact customers.
* **Inventory Control:** Automated stock level management that decreases when a product is sold.
* **Reporting:** Basic sales reports and analytics provided by WooCommerce.

**5. Non-Functional Requirements**

* **Security:**
  + All traffic will be served over **HTTPS** using an SSL/TLS certificate (provided by Azure App Service).
  + Payment processing will be handled by PCI-compliant gateways (Stripe/PayPal).
  + Regular updates to WordPress core, plugins, and themes are mandatory.
* **Performance:**
  + Target a page load time of under 3 seconds.
  + Utilize a caching plugin (e.g., W3 Total Cache) and offload media to Azure Blob Storage.
* **Scalability:**
  + Azure App Service can be configured to **auto-scale** (add more server instances) based on CPU or memory load to handle traffic spikes.
  + The Azure MySQL database can be scaled up independently as needed.
* **Backup & Recovery:**
  + Implement daily automated backups of both the website files (via App Service) and the database (via Azure Database for MySQL).

**6. Deployment & Maintenance Plan 🚀**

* **Initial Deployment:**
  1. Provision **Azure Database for MySQL**.
  2. Provision **Azure App Service for Linux** using the "WordPress on App Service" marketplace offering, connecting it to the created database.
  3. Configure DNS to point the custom domain to the App Service instance.
  4. Install and configure the WooCommerce plugin, theme, and payment gateways.
* **Ongoing Maintenance:**
  1. **Weekly:** Check for and apply updates to WordPress core, plugins, and themes in a staging environment before pushing to production.
  2. **Daily:** Monitor automated backups to ensure they are completing successfully.
  3. **As Needed:** Monitor site performance and security logs. Scale Azure resources up or down based on traffic trends.