**System\_Architecture:**

**1. Data Sources**

* **Customer Database**: Stores customer demographic information, such as age, gender, and location.
* **Transaction Data**: Contains purchase history, including product IDs, quantities, timestamps, and purchase amounts.
* **Product Catalog**: Provides product details like categories, prices, and descriptions.

**2. ETL (Extract, Transform, Load) Process**

* **Data Extraction**: Pull data from various sources (e.g., Customer Database, Transaction Data, Product Catalog).
* **Data Transformation**: Clean, preprocess, and transform data. This includes handling missing values, encoding categorical data, and standardizing numerical features.
* **Data Loading**: Load the processed data into a centralized **Data Warehouse** or **Data Lake** for easy access during model training.

**3. Model Training Pipeline**

* **Feature Engineering**: Generate features that capture customer behavior (e.g., average spend, frequency of purchases).
* **Segmentation Model**: Use clustering algorithms (e.g., K-Means) to segment customers based on behavior patterns.
* **Recommendation Model**: Implement collaborative filtering or content-based filtering to recommend products to each segment.

**4. Testing**:

* **Jenkins**: For continuous integration and automating the testing process.
* **PyTest**: To write and run tests for validating the codebase and model functionality.
* **Ansible**: Used for automated deployment of the application in testing environments.
* **MySQL**: For data verification, ensuring that the database interactions are correctly managed and the data integrity is maintained.
* **Environment**:
  + **Testing Environment**: Where initial tests are conducted to validate functionality.
  + **Staging Environment**: A replica of the production environment where further testing is done before final deployment.

**5. Production:**

* **Jenkins**: For automating the deployment pipeline and integrating code changes seamlessly.
* **Ansible**: Used to deploy the application in the production environment and ensure configuration consistency.
* **Docker (Optional)**: For containerization, allowing for easier deployment and scaling of the application.
* **MySQL**: Manages the production database, supporting data storage, and retrieval.
* **Python**: The primary language for application code, used to run the backend services and recommendation models.
* **Environment**:
  + **Production Environment**: The live environment where the application is deployed for end-users, with monitoring and logging tools in place for ongoing support and maintenance.

A diagram of a software development

Description automatically generated with medium confidence