**Project Summary Document**

**Group Members’ Names and IDs**

* **Shahrukh Kaleem**  
  ID:241828
* **Shahzaib Qamar**

**Project Description**

The project is an **E-commerce Store Database** designed to support a web-based shopping application named "Click Shopping." The database, named ecommerce\_store, manages essential e-commerce functionalities such as product catalog management, user accounts, shopping carts, and order processing. It consists of the following tables with their relationships enforced via foreign key constraints (default RESTRICT behavior):

* **Tables**:
  + site\_user: Stores user information (e.g., email, password, phone number).
  + product\_category: Manages product categories (e.g., Electronics, Food).
  + product: Contains product details (e.g., name, description, image, linked to a category).
  + product\_item: Stores individual product items with stock and pricing (linked to a product).
  + shopping\_cart: Represents a user’s cart (linked to a user).
  + shopping\_cart\_item: Manages items in a cart (linked to a cart and product item).
  + shipping\_method: Stores shipping options (e.g., Standard, Express).
  + order\_status: Defines order statuses (e.g., Pending, Shipped).
  + shop\_order: Records orders (linked to shipping method and status).
  + order\_line: Contains order items (linked to an order and product item).
* **Key Features**:
  + Add products with categories (addproducts.php).
  + Display products with pricing and discounts (products.php).
  + Manage shopping carts and orders.
  + Handle user authentication (implied login.php, register.php).
  + Support for image uploads for products (local uploads or URLs).
* **Purpose**: The database supports a fully functional e-commerce website, allowing users to browse products, add them to a cart, and place orders, while administrators can manage products, categories, and orders. Foreign key constraints ensure data integrity, requiring careful deletion of dependent records due to the RESTRICT behavior.

**Setup Instructions**

Follow these steps to set up the database, restore the schema, populate it with sample data, and run queries to test the application.

1. **Prerequisites**:
   * Install XAMPP (or a similar LAMP/WAMP stack) on your system.
   * Ensure MySQL and Apache are running (C:\xampp\xampp-control.exe).
   * Place your project files in the web server directory: C:\xampp\htdocs\Ecommerce\_store\.
2. **Create the Database**:
   * Open phpMyAdmin by navigating to http://localhost/phpmyadmin in your browser.
   * Create a new database named ecommerce\_store:

CREATE DATABASE ecommerce\_store;

* + Select the ecommerce\_store database from the left sidebar.

1. **Restore the Database Schema**:
   * Use the following SQL script to create all tables with foreign key constraints. Save it as create\_ecommerce\_store\_tables.sql in C:\xampp\htdocs\Ecommerce\_store\.

* In phpMyAdmin, go to the "Import" tab, select
* create\_ecommerce\_store\_tables.sql from repository
* and click "Go" to create the tables.
* Verify the tables are created:
* SHOW TABLES;.

1. **Populate the Database with Sample Data**:
   * Use the following script to insert sample data into the tables.
   * Save it as
   * populate\_ecommerce\_store.sql in C:\xampp\htdocs\Ecommerce\_store\.

populate\_ecommerce\_store.sql is in repository

* In phpMyAdmin, go to the "Import" tab, select populate\_ecommerce\_store.sql, and click "Go".
* Verify the data: SELECT \* FROM product; and SELECT \* FROM product\_item;.

1. **Set Up the Uploads Directory**:
   * Create an uploads/ directory in C:\xampp\htdocs\Ecommerce\_store\ to store uploaded product images:

mkdir C:\xampp\htdocs\Ecommerce\_store\uploads

* + Ensure the directory is writable by the web server (Apache). On Windows, this is typically automatic, but you can set permissions if needed:

icacls "C:\xampp\htdocs\Ecommerce\_store\uploads" /grant Everyone:F

* + If using local image paths (e.g., uploads/pizza.jpg), place sample images in this directory or adjust paths in the data.

1. **Run Sample Queries**:
   * Use the following script, which includes sample queries for insertion, retrieval, updates, and deletions. Save it as ecommerce\_store\_queries.sql in C:\xampp\htdocs\Ecommerce\_store\.

ecommerce\_store\_queries.sql

* In phpMyAdmin, go to the "Import" tab, select ecommerce\_store\_queries.sql, and click "Go" to run the queries.
* Alternatively, copy and execute specific sections (e.g., insertions) in the "SQL" tab.

1. **Test the Application**:
   * Ensure your PHP files (e.g., addproducts.php, products.php) are in C:\xampp\htdocs\Ecommerce\_store\.
   * Access the application via the browser:
     + http://localhost/Ecommerce\_store/addproducts.php to add new products.
     + http://localhost/Ecommerce\_store/products.php to view products.
   * Verify that products are displayed correctly and that you can add new products with valid category IDs.
2. **Troubleshooting**:
   * **Foreign Key Errors**: If deletions fail (e.g., DELETE FROM product\_category WHERE id = 3), update or delete child records first:

UPDATE product SET category\_id = NULL WHERE category\_id = 3;

DELETE FROM product\_category WHERE id = 3;

* + **Image Upload Issues**: If uploading images fails in addproducts.php, check php.ini settings (file\_uploads = On, upload\_max\_filesize = 5M) and restart Apache.
  + **No Products Displayed**: Ensure the product and product\_item tables are populated (SELECT \* FROM product;). If empty, rerun the population script.